

日 本 国 特 許 庁

PATENT OFFICE
JAPANESE GOVERNMENT

JC828 U.S. PTO
09/580601
05/25/00

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日

Date of Application:

1 9 9 9 年 1 2 月 2 8 日

出 願 番 号

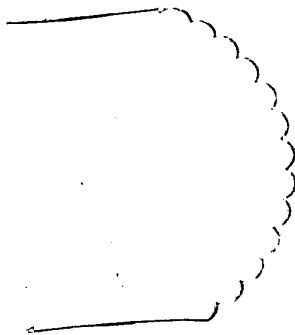
Application Number:

平成 1 1 年 特 許 願 第 3 7 2 6 0 6 号

出 願 人

Applicant (s):

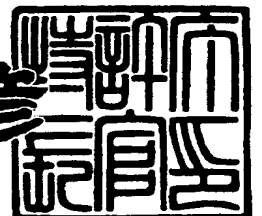
松下電器産業株式会社



2 0 0 0 年 3 月 1 7 日

特 許 庁 長 官
Commissioner,
Patent Office

近 藤 隆 彦



出 証 番 号 出 証 特 2 0 0 0 - 3 0 1 7 7 0 8

【書類名】 特許願

【整理番号】 2032710003

【提出日】 平成11年12月28日

【あて先】 特許庁長官 殿

【国際特許分類】 G11B 27/00

G11B 19/02

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 廣田 照人

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 田川 健二

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 石川 智一

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 井上 信治

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 松島 秀樹

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 小塚 雅之

【特許出願人】

【識別番号】 000005821

【氏名又は名称】 松下電器産業株式会社

【代理人】

【識別番号】 100090446

【弁理士】

【氏名又は名称】 中島 司朗

【選任した代理人】

【識別番号】 100109210

【弁理士】

【氏名又は名称】 新居 広守

【先の出願に基づく優先権主張】

【出願番号】 平成11年特許願第149893号

【出願日】 平成11年 5月28日

【先の出願に基づく優先権主張】

【出願番号】 平成11年特許願第236724号

【出願日】 平成11年 8月24日

【手数料の表示】

【予納台帳番号】 014823

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9810105

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 半導体メモリカード、その再生装置及び記録装置

【特許請求の範囲】

【請求項 1】 オーディオストリームを分割することにより得られた複数のオーディオブロックであって、各々が互いに異なる暗号鍵にて暗号化されているものを収録しているファイルからなるファイル群と、

前記各ファイルに収録されているオーディオブロックについての諸特性を示す管理情報をテーブル形式に配してなる管理情報テーブルと、

前記各オーディオブロックをどのような順序で再生させるかを示すプレイリスト情報と

を格納していることを特徴とする半導体メモリカード。

【請求項 2】 前記オーディオストリームは、複数のコンテンツを含むものであり、

複数のコンテンツのうち、その再生が所定の上限時間以内に完遂するものは、1つのオーディオブロックとして1つのファイルに収録されていて、

再生が所定の上限時間以内に完遂しない残りのコンテンツは、複数のオーディオブロックに分割され、同数のファイルに収録されており、

前記管理情報テーブルに格納されている各ファイルについての管理情報は、

ファイルに収録されているオーディオブロックが、コンテンツそのものであるか、コンテンツの一部であってその先頭部分であるか、中間部分であるか、終端部分であるかを示す第 1 属性情報を含む

ことを特徴とする請求項 1 記載の半導体メモリカード。

【請求項 3】 オーディオブロックを収録している複数のファイルのうち、何れかのものが削除されれば、

削除されたファイルに対応づけられている管理情報における第 1 属性情報は、当該オーディオブロックが、削除済みであることを示す値に書き換えられる

ことを特徴とする請求項 2 記載の半導体メモリカード。

【請求項 4】 前記プレイリスト情報は、

前記オーディオブロックのそれぞれを指し示す複数のサーチポイントからなり

、各サーチポイントの並びにより、オーディオブロックの再生順序を規定するものであり、

前記各サーチポイントは、

ファイルに収録されているオーディオブロックが、コンテンツそのものであるか、コンテンツの一部であってその先頭部分であるか、中間部分であるか、終端部分であるかを示す第2属性情報を含む

ことを特徴とする請求項2又は3記載の半導体メモリカード。

【請求項5】 オーディオブロックを収録している複数のファイルのうち、何れかのものが削除されれば、

プレイリスト情報において、当該オーディオブロックを指し示すサーチポイントが含まれている第2属性情報は、当該オーディオブロックが、削除済みであることを示す値に書き換えられる

ことを特徴とする請求項4記載の半導体メモリカード。

【請求項6】 前記所定の上限時間は、間欠的な再生時における一スキップ分の時間を m 倍した時間であり(m は、2以上の整数)、

前記管理情報は更に、

オーディオブロックの内部を前記間欠的に再生する際、スキップ先となるアドレスを示す m 個以下のエン트리情報を含む

ことを特徴とする請求項2～5の何れかに記載の半導体メモリカード。

【請求項7】 前記オーディオブロックは、 m 個以下のオーディオエレメントを含み、

管理情報におけるエン트리情報は、各オーディオエレメントの先頭位置を示す

ことを特徴とする請求項6記載の半導体メモリカード。

【請求項8】 前記管理情報は更に、ブロック情報テーブルを含み ブロック情報テーブルは、

オーディオブロックに含まれるオーディオエレメントのうち、その先頭のものに、復号可能な最小単位であるオーディオフレームがどれだけ含まれているかを示す先頭フレーム数情報、

オーディオブロックに含まれるオーディオエレメントのうち、その終端のものに、オーディオフレームがどれだけ含まれているかを示す終端フレーム数情報、

オーディオブロックに含まれるオーディオエレメントのうち、その中間のものに、オーディオフレームがどれだけ含まれているかを示す中間フレーム数情報を含む

ことを特徴とする請求項7記載の半導体メモリカード。

【請求項9】 前記ブロック情報テーブルは更に、

複数のファイルのそれぞれにおいて、ファイル先頭から、それに収録されているオーディオブロックの先頭まで無効部分のデータ長を示すデータオフセット情報、オーディオブロックの末尾からそれを収録しているファイルの末尾までの間に無効領域が存在する場合に、オーディオブロックの実効データ長を示す実効データ長情報を含む

ことを特徴とする請求項8記載の半導体メモリカード。

【請求項10】 半導体メモリカードについての記録装置であって、

装置外部から入力されてくる入力信号に従って、オーディオストリームを順次生成する生成手段と、

生成されたオーディオストリームを順次蓄積するバッファと、

半導体メモリカードにファイルを作成して、バッファに蓄積されたオーディオストリームをファイルに順次書き込む第1書込手段と、

復号可能な最小単位であるオーディオフレームが、ファイル内にどれだけ書き込まれたかを示すフレーム数を計数する計数手段と、

フレーム数の計数値が所定数に達する度に、当該所定数のオーディオフレームから構成されるオーディオエレメントの先頭位置を示すエントリー情報を生成する生成手段と、

生成されたエントリー情報の数を計数する計数手段と、

エントリー情報の計数値が所定数に達する度に、第1書込手段に別のファイルを作成させて、バッファに蓄積されたオーディオストリームを作成された当該別ファイルに順次書き込ませる制御手段と、

エントリー情報の計数値が所定数に達する度に、所定数のエントリー情報から

なる複数のエン트리情報を含む管理情報を半導体メモリカードに書き込む第2書込手段と

を備えることを特徴とする記録装置。

【請求項 1 1】 請求項 1 記載の半導体メモリカードについての再生装置であって、

半導体メモリカードからプレイリスト情報を読み出す読出手段と、

読み出されたプレイリスト情報により、再生順序が指定されている複数のオーディオブロックの何れかを再生対象として特定する特定手段と、

再生対象として特定されたオーディオブロックを含むファイルを半導体メモリカードから読み出す第2読出手段と、

第2読出手段により読み出されたファイルに含まれるオーディオブロックから、デコード可能な最小単位であるオーディオフレームを抽出して順次出力する出力手段と、

再生対象として特定されたオーディオブロックについての暗号鍵を用いて、出力されたオーディオフレームの暗号化を解除する解除手段と、

暗号化が解除されたオーディオフレームをデコードして、音声再生を行うデコード手段とを備え、

前記特定手段は、

読み出されたオーディオブロックに含まれるオーディオフレームが全て出力されれば、プレイリスト情報に示されている再生順序において当該オーディオブロックの次順位に位置付けられているオーディオブロックを再生対象として特定して、当該新たな再生対象について、第2読出手段、出力手段、解除手段、デコード手段に同様の処理を行わせる

ことを特徴とする再生装置。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、オーディオデータ、制御データを格納する半導体メモリカードと、当該半導体メモリカードについての再生装置とに関し、特に、電子音楽配信等の

コンテンツ配信サービスにおいて、コンテンツとして配信されたオーディオデータ、制御データを格納する場合の改良に関する。

【 0 0 0 2 】

【従来の技術】 近年、電子音楽配信を実現するためのインフラストラクチャが整備されつつあり、CD、カセットテープ等のパッケージソフトの流通を主体としていた音楽業界は、壮大な変革期を迎えようとしている。

かかる電子音楽配信における音楽コンテンツの代金決済は、音楽会社のサーバコンピュータと、消費者が所有するコンピュータとの間で、電子商取引に準じた手続きを経ることにより行われる。また、音楽コンテンツの引き渡しは、消費者が所有するコンピュータが、音楽会社のサーバコンピュータから音楽コンテンツをダウンロードすることにより実現される。電子音楽配信の利用者は、そのような手続きで入手した音楽コンテンツを持ち運ぼうとする場合、音楽コンテンツを可搬式の記録媒体に格納させねばならない。電子音楽配信により入手した音楽コンテンツの格納に最も適しているといわれているものに半導体メモリカードがある。

【 0 0 0 3 】

半導体メモリカードは、フラッシュATAカードやコンパクトフラッシュカードと呼ばれるものが知られている。これら半導体メモリカードは、フラッシュメモリ (EEPROM) と呼ばれる半導体デバイスを内蔵しており、MD、CD-R等と比較して、データの高速書き込みが可能であり、膨大なデータ長を有する音楽コンテンツの格納を短時間で完遂することができる。フラッシュメモリは、記憶内容の維持のための電力供給が不要であり、コンパクトディスク (CD)、ミニディスク (MD) についての再生装置と比較して、ディスク板体やヘッドを駆動するための駆動機構や駆動電力が不要であるので、CD、MDの再生装置より一層の小型化・省電力化を実現することができる。更にCD、MDの再生装置は持ち運び時に振動が発生した場合音飛びが発生するが、半導体メモリカードではそのような音飛びが発生しないので、半導体メモリカードの再生装置は、持ち運びながら音楽を鑑賞するという携帯用途に最適である。

【 0 0 0 4 】

その反面、電子音楽配信にて配信された音楽コンテンツが半導体メモリカードに格納されれば、半導体メモリカードに格納された音楽コンテンツを更にコピーすることにより、音楽著作物の不正な複製品が氾濫する恐れがある。特に半導体メモリカードは、CD-R,MDと比較して、高速なデータ書き込みが可能であるので、複製品の氾濫は、より深刻化すると考えられる。そのような著作権者の不安を払拭するためには、電子音楽配信にて配信された音楽コンテンツを半導体メモリカードに格納する場合に、著作権者側の不安を払拭するような高度な暗号化方式で、音楽コンテンツを暗号化せねばならない。不正コピーを防止する方法の1つに、特開平5-257816公報に記載されている暗号化方法がある。この暗号化方法は、音楽コンテンツを記録媒体固有の識別情報で暗号化して格納するというものであり、記録媒体固有の識別情報で音楽コンテンツが暗号化されれば、その記録媒体に格納された音楽コンテンツは、正当に再生されるが、異なる記録媒体に格納された音楽コンテンツの複製品は、音楽コンテンツの再生が不可能となるので、音楽コンテンツの二次コピーを防止することができる。

【0005】

また磁気式、光学式の記録媒体と比較して、半導体メモリカードはその容量が制限されていることが多いので、音楽コンテンツを半導体メモリカードに格納する際には、音楽コンテンツをより高度に圧縮しておく必要がある。そのような高圧縮率を得ることができる符号化方式の1つにMPEG2-AAC方式がある。MPEG2-AAC方式の特徴は、人間の聴覚特性を考慮して、各オーディオフィレーム（オーディオフィレームとは、符号化、再生の最小単位となる時間帯であり、大体20msecの時間長に相当する）に割り当てるべきビット長を変化させる点にある。即ち、可聴帯域（人間が聞くことができる帯域）を多く有するオーディオフィレームに多くのビット長を割り当て、不可聴帯域（人間が聞く事ができない帯域）を多く有するオーディオフィレームには、割り当てるべきビット長を削減するのである。

【0006】

このようにMPEG2-AAC方式は可聴帯域の含有量に応じてオーディオフィレームに割り当てるべきビット長を変化させるので、言い換えれば可変長符号化方式であるので、高圧縮率且つ高音質な音楽コンテンツを得ることができる。高圧縮率且

つ高音質な音楽コンテンツは、公衆回線での伝送に好適であり、容量が制限された半導体メモリカードに蓄積する場合に好適である。

【0007】

【発明が解決しようとする課題】

ところで、従来の暗号化方式では、その音楽コンテンツの暗号化に用いた暗号鍵が解読されれば、その記録媒体に記録されている全ての音楽コンテンツの復号が可能になるので、いったん暗号鍵が暴露されると、その半導体メモリカードに格納されている全ての曲が容易に復号化されてしまうという第1の問題点がある。そのような事態は頻繁に起こるとは言い難いが、希にでも発生すれば、それに起因して生じる著作権者の損害は、はかりしえないものとなる。特に、近年におけるコンピュータの処理能力の向上には目覚ましいものがあり、音楽コンテンツの暗号化に用いた暗号鍵が解読されることは、完全に有り得ないとは断言し難い。万が一にでも暗号鍵が暴露された場合に、著作権者の損害を最小限に留めるようなデータ構造は、従来技術には存在しないという問題点がある。

【0008】

また、電子音楽配信において配信されるべき音楽コンテンツは、著作権を保護せねばならないので、暗号化された状態のまま音楽コンテンツを配信し、半導体メモリカードも、暗号化がなされたまま、音楽コンテンツを格納しておく必要がある。しかし暗号化された状態のまま音楽コンテンツを半導体メモリカードに格納した場合、正当な対価を支払って音楽コンテンツを入手した操作者から、音楽コンテンツを自由に編集する機会を奪ってしまうという第2の問題点がある

即ち、音楽コンテンツが暗号化されれば、曲の再生順序を入れ替えたり、また一部の曲を削除するという編集が困難となる。これは、電子音楽配信において正当な対価を支払って音楽コンテンツを入手した操作者からも音楽コンテンツを自由に編集する機会を奪う結果になり、著作権保護が過剰化してしまうという問題点がある。

【0009】

一方、半導体メモリカードと同じ音楽の録音用途に用いられるミニディスク(MD)の記録装置は、再生順序の入れ替えや、曲の分割、統合等の編集処理が可能で

あるので、半導体メモリカードにおいて編集機能が実現できないのでは、既存のミニディスクと比較して、機能的に見劣りすることになってしまい、現状のミニディスクのユーザに、半導体メモリカードの機能価値を強くアピールできないという問題点がある。

【0010】

更に従来技術には、MPEG2-AAC等の可変符号長の符号化方式で符号化されている音楽コンテンツに対して特殊再生を行おうとすると、再生装置に、より大規模なメモリの実装を求めることとなり、再生装置の製造コストの高騰を招いてしまうという第3の問題点がある。ここで特殊再生とは、既存のMD,CDの再生装置で実現されている機能であって、音楽の再生時の頭出し（再生位置の特定）を容易にするものであり、間欠的な再生を繰り返しながら、倍速再生を行う順方向サーチ再生、逆方向サーチ再生、操作者から時刻の指定操作を受け付けて、指定された時刻から再生を開始するタイムサーチ機能といったものが広く用いられている。既存のMD,CDの再生装置との機能互換を果たすには、そのような特殊再生を半導体メモリカードの再生装置において実現することは、必要不可欠であると言える。ここで、音楽コンテンツが固定符号長の符号化方式で符号化されている場合、1分先、2分先の頭出し位置がどこであるかを特定するには、単位再生時間長のデータサイズの整数倍先のアドレスを参照すればよい。しかし音楽コンテンツがMPEG2-AAC等の可変符号長の符号化方式で符号化されている場合、1分先、2分先の頭出し位置が、単位再生時間長のデータサイズの非整数倍になり得るので、1分先、2分先のアドレスがどこであるかを予め記載したタイムサーチテーブルを併用する必要がある。ここで、演奏時間が短い曲については、タイムサーチテーブルに記述しておく再生位置の数は少なく済むが、その演奏時間が長い曲については、タイムサーチテーブルに多くの再生位置を記述しておく必要があり、多くの再生位置の記述が必要ならば、タイムサーチテーブルは大規模となる。一方、再生装置は、上記のような特殊再生を行う場合、タイムサーチテーブルをメモリ上に配置して、アクセスせねばならないが、再生時間が長い曲についてのタイムサーチテーブルが大きなデータサイズになることを想定すると、それを格納しておくメモリの規模も大規模ならざるを得ない。そのように、メモリの規模

が大規模になるのなら、再生装置の製造コストを高騰を招いてしまうという問題点がある。

【 0 0 1 1 】

本発明の第 1 の目的は、音楽コンテンツの著作権を保護しつつも、音楽コンテンツの編集を行うことができる半導体メモリカードを提供することである。

本発明の第 2 の目的は、ハードウェアの処理性能の向上を再生装置に求めなくても音楽著作物の著作権を保護することができ、かつ、再生装置に実装されているメモリ規模が小規模であっても、早送り、巻戻し等の特殊再生を行うことができる半導体メモリカードを提供することである。

【 0 0 1 2 】

【課題を解決するための手段】

本発明の第 1 の目的は、オーディオストリームを分割することにより得られた複数のオーディオブロックであって、各々が互いに異なる暗号鍵にて暗号化されているものを収録しているファイルからなるファイル群と、前記各ファイルに収録されているオーディオブロックについての諸特性を示す管理情報をテーブル形式に配してなる管理情報テーブルと、前記各オーディオブロックをどのような順序で再生させるかを示すプレイリスト情報とを格納していることを特徴とする半導体メモリカードにより達成される。

【 0 0 1 3 】

本発明の第 2 の目的は、上述した半導体メモリカードの前記管理情報に、オーディオブロックの内部を間欠的に再生する際、スキップ先となるアドレスを示す 1 個以下のエン트리情報を含ませることにより達成される。

【 0 0 1 4 】

【発明の実施の形態】

以降、図面を参照しながら半導体メモリカード（フラッシュメモリカード）の実施形態について説明を行う。

尚、以降の各文には、その文頭に以下のような体系を有する分類番号を付している。

【 0 0 1 5 】

{x1-x2_x3-x4}

分類番号の桁数は、その項目の階層的な深さを意味している。具体的にいうと、x1は、説明に引用している図番である。本明細書に添付している図には、明細書において引用する順番に沿った番号を付しているので、この図番の順序が、説明の順序とほぼ同一となる。x2は、x1に示される図を引用して説明する場合の説明の順序を示す。x3は、x2の構成要素をより詳細に説明するために説明図を引用する場合、その説明図の図番を示し、x4は、x3に示される図を引用して説明する場合の説明の順序を示す。

【 0 0 1 6 】

(第 1 実施形態)

{1-1_2} フラッシュメモリカード 3 1 の外観形状

初めに、フラッシュメモリカード 3 1 の外観形状について説明する。図 1 は、フラッシュメモリカード 3 1 を上面から見た場合の形状示す図であり、図 2 は、フラッシュメモリカード 3 1 をその下面から見た場合の構造を示す図である。図 1、図 2 に示すように、フラッシュメモリカード 3 1 の大きさは、長さが約 32.0 mm、幅は約 24.0 mm、厚さ約 2.1 mm であり、指先で把持できる程度の大きさ（切手サイズの大きさ）である。下面には、機器との接続のための 9 本のコネクタが設けられており、側面には、記憶内容の上書きを許可するか禁止するかを操作者が設定することができるプロテクトスイッチ 3 2 が設けられている。

【 0 0 1 7 】

{3-1} フラッシュメモリカード 3 1 の物理構造

図 3 は、本実施形態に係る半導体メモリカード（以下、フラッシュメモリカード 3 1 と称する）の階層構造を示す図である。本図に示すように、フラッシュメモリカード 3 1 の階層構造は、物理層、ファイルシステム層、応用層からなる点で、DVD (Digital Video Disc) の階層構造と同一であるが、各層における論理構造、物理構造は大きく相違する。

【 0 0 1 8 】

{3-2} フラッシュメモリカード 3 1 の物理構造

先ずフラッシュメモリカード 3 1 の物理層について説明する。フラッシュメモ

りは、複数のセクタからなり、各セクタは512バイトのデジタルデータを格納する。例えば64MByteタイプのフラッシュメモリカード31の場合、そのメモリー容量は、 $67108864 (= 64 \times 1024 \times 1024)$ バイトであり、このときの有効セクタ数は $131072 (= 67108864 / 512)$ となる。更に、この有効セクタからエラー用の代替セクタ数を差し引けば、残りの有効セクタ数は、128,000となり、ここに各種データが記録されることとなる。

【0019】

{3-2_4(a)-1} 物理層における3つの領域

これら有効セクタからなる領域には、図4(a)に示す3つの領域が設けられる。図4(a)は、フラッシュメモリカード31の物理層に設けられた『特殊領域』、『認証領域』、『ユーザ領域』を示す図である。以降、これら3つの領域について説明する。

【0020】

『ユーザ領域』は、フラッシュメモリカード31と接続された機器が様々なデータを自由に書き込むことができ、データを自由に読み出すことができる領域であり、その内部領域がファイルシステムにより管理されている。

『特殊領域』は、フラッシュメモリカード31のそれぞれについてユニークな値を持つメディアIDが格納される領域である。ユーザ領域が書込可能であるのに対して、特殊領域は、読出専用であり、ここに格納されたメディアIDを書き換えることはできない。

【0021】

『認証領域』は、ユーザ領域同様、データ書き込みが可能な領域である。ユーザ領域との差違は、ユーザ領域では、データの読み書きが自由に行なえるのに対して、認証領域では、フラッシュメモリカード31と接続された機器と、フラッシュメモリカード31とが互いの正当性を確認した場合のみ読み書きすることができる点、即ち、フラッシュメモリカード31と接続された機器と、フラッシュメモリカード31との相互認証が成功した場合のみ、読み書き可能となる点である。

【0022】

{3-2_4(a)-2} 物理層における3つの領域の用途

フラッシュメモリカード 31 に接続された機器がフラッシュメモリカード 31 にデータを書き込む際、そのデータの著作権保護の要否に応じて、これら3つの領域は利用される。ここで、著作権の保護が必要なデータをフラッシュメモリカード 31 に書き込む場合、当該データは、所定の暗号鍵を用いて暗号化された後にユーザ領域に格納される。この暗号鍵は著作権者が自由に設定できるものであり、これだけでも、当該データの著作権は保護されるが、更に万全を期すため、この暗号化に用いた暗号鍵自身も暗号化する。暗号鍵自身を暗号化する際、暗号鍵として用いられるのは、特殊領域に格納されているメディアIDを所定の演算式に適用することにより得られる任意の値であり、認証領域は、当該任意の値を用いて暗号化された暗号鍵を格納する。著作権保護が必要なデータは、所定の暗号鍵を用いて暗号化し、この暗号鍵自身もメディアIDに基づいた値を用いて暗号化するという二段階の暗号化がなされるので、不正コピーなどの著作権侵害行為は、極めて困難になる。

【0023】

{3-2_4(b)-1} ファイルシステムの概要

フラッシュメモリカード 31 の物理層の構成は以上説明した通りであり、著作権保護の改良がなされていることがわかる。続いてこの物理層上に存在するファイルシステム層の構成について説明する。

DVDのファイルシステム層は、UDF(universal disk format)型のファイルシステムであるの対して、フラッシュメモリカード 31 のファイルシステム層は、FAT型のファイルシステム (FAT:File Allocation Table,ISO/IEC 9293) であり、この点がDVDと異なる。

【0024】

図4 (b) は、ファイルシステム層における認証領域及びユーザ領域の構成を示す図である。図4 (b) においてファイルシステムにおける認証領域及びユーザ領域は、『パーティションブートセクタ』と、『ファイルアロケーションテーブル(FAT)』と、『ルートディレクトリエントリ』と、『データ領域』とを含んでおり、認証領域とユーザ領域は共に同じ構成となっていることがこの図からも

明らかである。図5は、これらファイルシステム構成の詳細を示す図である。以降、ユーザ領域についての構成を図4、図5を参照しながら説明する。

【0025】

{3-2_4(b)-2} パーティションブートセクタ

『パーティションブートセクタ』は、フラッシュメモリカード31が汎用パーソナルコンピュータに装填され、当該汎用パーソナルコンピュータのオペレーティングシステムの起動ディスクにフラッシュメモリカード31を割り当てられた場合、汎用パーソナルコンピュータがブート時に参照すべき内容が記載されているセクタである。

【0026】

{3-2_4(b)-3_5} データ領域

『データ領域』は、クラスタを最小単位にして、フラッシュメモリカード31に接続された機器によりアクセスされる領域である。フラッシュメモリカード31のセクタサイズが512バイトであるのに対して、クラスタサイズは、16Kバイトであるので、ファイルシステム層では32個のセクタを一単位として、データの読み書きが行われる。クラスタサイズを16Kバイトとした理由は、以下の通りである。即ち、フラッシュメモリカード31にデータを書き込む場合、当該フラッシュメモリカード31に格納されているデータを一旦イレーズ(消去)してから、データ書き込みを行わねばならない。フラッシュメモリカード31において、そのようにデータをイレーズできるサイズは、16Kバイトであるので、このイレーズ可能なサイズにクラスタサイズを設定することにより、データ書き込みが好適に行われるようにしている。図5における破線の引き出し線ff2は、データ領域に含まれる複数のクラスタ002,003,004,005……を示す。図中の番号002,003,004,005,006,007,008……は、各クラスタを識別するために付与された3桁の16進数表記のクラスタ番号を示す。データ領域に対するアクセスは、クラスタを最小単位として行われるので、データ領域の内部位置は、これらのクラスタ番号を用いて、指示される。

【0027】

{3-2_4(b)-4_5} ファイルアロケーションシステム

『ファイルアロケーションシステム』は、ISO/IEC 9293に準拠したファイルシステム構造を有しており、複数のFAT値からなる。各FAT値は各クラスタに対応づけられており、対応するクラスタが読み出された場合、次にどのクラスタを読み出さばよいかを示す。図5の破線の引き出し線ff1は、ファイルアロケーションテーブルに含まれる複数のFAT値002,003,004,005…を示す。このFAT値に付与された数値『002,003,004,005…』は、各FAT値がどのクラスタに対応づけられているか、つまり、各FAT値が対応づけられているクラスタのクラスタ番号を示す。

【0028】

{3-2_4(b)-5_5-1} ルートディレクトリエントリ

『ルートディレクトリエントリ』は、ルートディレクトリにどのようなファイルが存在するかを示す情報である。具体的にいうと、ルートディレクトリエントリーには、存在するファイルの『ファイル名』と、そのファイルの『拡張子』と、『ファイル属性』と、ファイルの『更新時刻及び年月日』と、ファイルの先頭部が格納されている『ファイル最初のクラスタ番号』とが記載されている。

【0029】

{3-2_4(b)-5_5-2} サブディレクトリのディレクトリエントリ

ルートディレクトリについての情報は、このルートディレクトリエントリーに記載されるが、サブディレクトリについての情報は、このルートディレクトリエントリーには記載されない。サブディレクトリについてのディレクトリエントリーは、データ領域内に作成される。図5のデータ領域内に記載されたSD-Audioディレクトリエントリーは、サブディレクトリについてのディレクトリエントリーの一例であり、本SD-Audioディレクトリエントリーは、ルートディレクトリエントリー同様、そのサブディレクトリに存在するファイルの『ファイル名』と、そのファイルの『拡張子』と、『ファイル属性』と、ファイルの『更新時刻及び年月日』と、ファイルの先頭部が格納されている『ファイル最初のクラスタ番号』とが記述される。

【0030】

{3-2_4(b)-5_6-1} AOBファイルの格納方式

ここで、SD-AudioディレクトリにAOB001.SA1というファイルを格納する場合、

AOB001.SA1がどのように格納されるか、即ち、ファイル格納方式の一例を図6を参照しながら説明する。上述したようにデータ領域の最小アクセス単位はクラスタであるので、AOB001.SA1は、クラスタサイズを最小単位にしてデータ領域に格納せねばならない。AOB001.SA1は、先ずクラスタサイズに分割されて、各クラスタに書き込まれる。図6は、AOB001.SA1をクラスタサイズに合わせて5つに分割し、各分割部分を、クラスタ003,004,005,00A,00Cに格納する状態を想定した図である。

【 0 0 3 1 】

{3-2_4(b)-5_7-1} AOBファイルの格納方式

AOB001.SA1が分割格納されると、ディレクトリエントリー及びファイルアロケーションテーブルは、図7のように設定されねばならない。

図7は、AOB001.SA1が複数のクラスタに記録されている場合のディレクトリエントリー及びファイルアロケーションテーブルについての設定例を示す図である。本図においてAOB001.SA1の先頭部分がクラスタ003に記録されている場合、SD-Audioディレクトリエントリーにおける『最初のクラスタ番号』には、その先頭部分が格納されているクラスタについてのクラスタ番号003が記載される。以降、AOB001.SA1の後続する部分は、クラスタ004、クラスタ005に格納されていることがわかる。AOB001.SA1の先頭部分を格納しているクラスタ003には、FAT値003(004)が対応しているが、このFAT値は、AOBファイルの後続する部分を格納しているクラスタ004を示すものである。またこれに後続している部分を格納しているクラスタ004,005には、FAT値004(005),FAT値005(00A)が対応しているが、このFAT値は、AOBファイルの次の後続する部分を格納しているクラスタ005,00Aを示すものである。

【 0 0 3 2 】

これらFAT値に記載されたクラスタ番号を矢印fk1,fk2,fk3,fk4,fk5……に示すように順次読みとってゆけば、AOB001.SA1の分割部分を全て読み取ることができる。以上の説明により、フラッシュメモリカード31のデータ領域は、クラスタを最小単位としてアクセスされ、また各クラスタにはそれぞれFAT値が対応づけられていることがわかる。尚、AOBファイルの末尾の部分を格納したクラスタ(

図 7 の一例では、クラスタ 00C) に対応づけられている FAT 値には、そのクラスタがファイルの最終部分を格納していることを示すクラスタ番号『FFF』が記述される。

【 0 0 3 3 】

以上で、本発明のフラッシュメモリカード 3 1 のファイルシステムに関する説明を終え、続いて、上述したファイルシステム上に存在する応用層の構成について説明する。

{3-3} フラッシュメモリカード 3 1 における応用層の概要

フラッシュメモリカード 3 1 における応用層の概要は、図 3 に記載された通りである。図 3 における破線の引き出し線 PN1 に示すようにフラッシュメモリカード 3 1 における応用層は、プレゼンテーションデータと、プレゼンテーションデータの再生を制御するためのナビゲーションデータとからなる。

【 0 0 3 4 】

本図の破線の引き出し線 PN2 に示すように、プレゼンテーションデータは、音楽等の音声データをエンコードすることにより得られたオーディオオブジェクト群(AOB群)を含み、ナビゲーションデータは、プレイリストマネージャー(PlaylistManager(PLMG))と、トラックマネージャー(Track Manager(TKMG))とを含む。

{3-3_8(a)(b)-1} ディレクトリ構成

図 8 (a) (b) は、応用層におけるこれら 2 つのデータを格納する場合、ファイルシステム層においてユーザ領域及び認証領域には、どのようなディレクトリが構成され、どのようなファイルが当該ディレクトリの配下に作成されるかを示す図である。本図における『SD_AUDIO.PLM』、『SD_AUDIO.TKM』は、プレイリストマネージャー(PlaylistManager(PLMG))、トラックマネージャー(Track Manager(TKMG))といったナビゲーションデータを収録したファイルであり、『AOB001.SA1』『AOB002.SA1』『AOB003.SA1』『AOB004.SA1』……は、プレゼンテーションデータであるオーディオオブジェクトを格納したファイル(以下、AOBファイルという)である。

【 0 0 3 5 】

『AOB0xx.SA1』における拡張子『SA』は、『Secure Audio』の略であり、これ

らの格納内容は、著作権保護の必要性があることを示す（尚、図 8（a）には AOB ファイルが 8 個だけ記述されているが、これは単なる一例であり、SD-Audio ディレクトリは AOB ファイルを最大 999 個まで格納することができる。）。このようにプレゼンテーションデータに著作権保護の必要性がある場合、認証領域には、SD-Audio ディレクトリという名称のサブディレクトリが設けられ、その SD-Audio ディレクトリの配下に SD-Audio の下に暗号鍵格納ファイル AOB SA1.KEY が作成される。図 8（b）は、SD-Audio の下に格納された暗号鍵格納ファイル AOB SA1.KEY を示す図である。

【 0 0 3 6 】

電子音楽配信において音楽会社のサーバコンピュータは、この図 8（a）（b）に示す SD-Audio ディレクトリを保持しており、当該音楽コンテンツの購入要求が消費者から発せられれば、この SD-Audio ディレクトリを圧縮し、暗号化した後、購入要求を発した消費者が所有する SD-Audio ディレクトリを公衆回線を介して送信する。消費者が所有するコンピュータがこの SD-Audio ディレクトリを受信すると、このディレクトリの暗号化を解除すると共に、伸長を行い、SD-Audio ディレクトリを得る（尚、ここでいう公衆回線は、ISDN 回線等の有線通信網、携帯電話に代表される無線通信網等、公衆に利用が解放されている全てのネットワークを含む）。尚、AOB ファイルを音楽会社のサーバコンピュータからダウンロードし、消費者が所有するコンピュータが、フラッシュメモリカード 3 1 においてこの図 8（a）（b）に示す SD-Audio ディレクトリを作成しても良い。

【 0 0 3 7 】

{3-3_9-1} AOB SA1.KEY と、AOB ファイルとの対応

図 9 は、SD-Audio の下にある AOB SA1.KEY と、AOB ファイルとの対応を示す図である。本図においてユーザ領域における暗号化ファイルを暗号化する際に用いた暗号鍵は、認証領域に対応する暗号鍵格納ファイルに格納される。

暗号化された AOB ファイルと、暗号鍵格納ファイルとは、以下の一定の規則 (1) (2) (3) に基づく対応関係を有する。

【 0 0 3 8 】

(1) 暗号鍵格納ファイルは、暗号化されたファイルが格納されているディレク

トリと同じディレクトリ名に配置される。図9のユーザ領域においてSD-AudioディレクトリにAOBファイルが配されており、暗号鍵格納ファイルもSD-Audioディレクトリに配されていることから、この規則に従った、ファイル配置が行われていることがわかる。

【0039】

(2)暗号鍵格納ファイルには、データ領域におけるAOBファイルのファイル名の先頭3文字と、所定の拡張子「.key」とを組み合わせたファイル名が付与される。AOBファイルのファイル名が『AOB001.SA1』である場合、暗号鍵格納ファイルには、矢印nk1,nk2に示すように、この先頭3文字『AOB』と、『SA1』と、拡張子『.key』とからなる『AOBSA1.KEY』というファイル名が付与されることがわかる。尚、図9には、『AOBSA1.KEY』の他に、『AOB』と、『SA1』と、拡張子『.BUP』とからなる『AOBSA1.BUP』が存在することがわかる。これは、『AOBSA1.KEY』と同一のバックアップファイルであり、『AOBSA1.KEY』が破損して、暗号鍵の利用が不可能となった場合に用いられる。

【0040】

(3)AOBファイルのファイル名には、“001”,“002”,“003”,“004”といったシリアル番号が付与され、認証領域の暗号鍵格納ファイル内に設けられている『File Key Entry』にも同様のシリアル番号#1,#2,#3……#8が付与されている。各AOBファイルを暗号化する際に用いた暗号鍵は、同一のシリアル番号を有する『File Key Entry』に格納される。図9における矢印AK1,AK2,AK3は、AOBファイルと暗号鍵との対応関係を示す。即ち、ユーザ領域におけるAOB001.SA1は『File Key Entry#1』に格納されている暗号鍵と対応しており、AOB002.SA1は、『File Key Entry#2』以降に格納された暗号鍵、AOB003.SA1は『File Key Entry#3』以降に格納された暗号鍵に対応していることを示す。以上の(3)からもわかるように、AOBファイルの暗号化に用いた暗号鍵は、各ファイル毎に異なるものであり、それらは、ファイル名に組み込まれている“001”,“002”,“003”,“004”といったシリアル番号と、同一のシリアル番号を有する『File Key Entry』に格納されている。各AOBファイルを異なる暗号鍵を用いて暗号化されるので、仮に、特定のAOBファイルの暗号化キーが暴露された場合でも、他のAOBファイルは、暴露された暗号鍵を

用いても暗号化を解除することはできない。これにより、AOBファイルを暗号化した際の暗号鍵が暴露された場合の損害を最小限に留めることができる。

【 0 0 4 1 】

{3-3_10-1} AOBファイルの内部構成

続いてAOBファイルの内部構成について説明する。図 1 0 は、AOBファイルのデータ構成を階層的に示す図である。本図の第 1 段目は、AOBファイルを示し、第 2 段目は、AOBを示す。第 3 段目は、AOB_BLOCKを示し、第 4 段目はAOB_ELEMENT、第 5 段目は、AOB_FRAMEを示す。

【 0 0 4 2 】

図 1 0 の第 5 段目における『AOB_FRAME』は、AOBを構成する最小単位であり、ADTSヘッダと、ADTS (Audio Data Transport Stream) 形式のオーディオデータとからなる。ADTS形式のオーディオデータは、MPEG2-AAC [Low Complexity Profile]にて符号化され、16Kbps～144Kbpsの伝送速度で再生されるストリームデータである（尚、既存のコンパクトディスクに記録されるPCMデータの伝送速度は1.5Mbpsであるので、PCMデータと比較して、一段と低いことがわかる。）。これらのAOB_FRAME列のデータ構造は、電子音楽配信にて配信されるオーディオデータトランスポートに含まれるオーディオフレーム列と同一である。即ち、AOB_FRAME列として格納されるべきオーディオデータトランスポートストリームは、MPEG2-AACにてエンコードされ、更に暗号化された状態で、公衆回線を伝送し、消費者宛に伝送される。AOBファイルは、そのように伝送されたオーディオデータトランスポートストリームを、AOB_FRAME列として分割して格納しているのである。

【 0 0 4 3 】

{3-3_10-1_11} MPEG2-AACについて

MPEG2-AACの詳細に関しては、ISO/IEC 13818-7:1997(E) Information technology - Generic coding of moving pictures and associated audio information - Part7 Advanced Audio Coding (AAC)を参照されたい。ここで注意すべきは、AOBは、ISO/IEC13818-7に記述されているパラメータ表を図 1 1 のように制限して適用されたMPEG2-AAC方式にて圧縮されている点である。図 1 1 は、ISO/IEC13

818-7に記述されているパラメータ表を示す図であり、Parameter欄と、Value欄と、Comment欄の内容を示すコメント欄とからなる。

【 0 0 4 4 】

パラメータ欄『profile』に対応するValue欄は、『01』と記述されており、IS 0/IEC 13838-7で規定されているLC-profileの制限が適用されていることを示す。

パラメータ欄『sampling_frequency#index』に対応するValue欄は、『0011,0100,0101,0110,0111,1000,1001,1010,1011』と記述されており、『48kHz,44.1kHz,32kHz,24kHz,22.05kHz,16kHz』といったサンプリング周波数が適用されていることを示す。

【 0 0 4 5 】

パラメータ欄『number_of_data_block_in_frame』に対応するValue欄は、『00』と記述されており、1header/1raw_data_blockに設定されていることを示す。

{3-3_10-2_12} AOB_FRAMEの構成

『AOB_FRAME』は、以上の制限下で符号化されたオーディオデータを含むが、AOB_FRAMEに含まれるオーディオデータのデータ長は、その再生時間が20ミリ秒となるデータに過ぎない。しかし、MPEG2-AAC方式は可変長符号化方式であるので、各AOB_FRAMEに含まれるオーディオデータのデータ長は、それぞれのAOB_FRAME毎に異なる。以下、図 1 2 を参照しながら、AOB_FRAMEの構成の詳細について説明する。本図の第 1 段目は、AOB_FRAMEの全体構成を示し、第 2 段目は、AOB_FRAMEのそれぞれの部位がどのように暗号化されているかを示す。この第 2 段目を参照すれば、ADTSヘッダは非暗号化部、即ち、暗号化がなされていないことがわかる。また、オーディオデータは、暗号化された部分と、非暗号化部分との双方を含む。暗号化部分は、8バイトの暗号化データを複数配したものである。8バイトの暗号化データは、64ビットの元データを56ビットの暗号鍵を用いて暗号化することにより生成されている。非暗号化部分は、そのように64ビット単位に暗号化が行われた際、64ビットに満たないために暗号化されずに残したものである。

【 0 0 4 6 】

第 3 段目は、非暗号化部分であるADTSヘッダの内容を示す図である。ADTSヘッ

ダは7バイトであり、12ビットの同期ワード(FFFと設定されている)と、同じAOB_FRAMEに含まれるオーディオデータのデータ長と、そのオーディオデータをエンコードする際のサンプリング周波数とが記載されている。

{3-3_10-3_13} AOB_FRAMEのバイト長設定

図 1 3 は、3つのAOB_FRAMEにおいて、それぞれのAOB_FRAMEにおけるオーディオデータのバイト長がどのように設定されるかを示す図である。本図において、AOB_FRAME#1に含まれるオーディオデータ#1のデータ長はx1、AOB_FRAME#2に含まれるオーディオデータ#2のデータ長はx2、AOB_FRAME#3に含まれるオーディオデータ#3のデータ長はx3であり、x1,x2,x3は、それぞれのデータ長が互いに異なる場合、AOB_FRAME#1に含まれるADTSヘッダには、データ長x1が記載され、AOB_FRAME#2に含まれるADTSヘッダには、データ長x2、AOB_FRAME#3に含まれるADTSヘッダには、データ長x3が記載される。オーディオデータそのものは、暗号化されているが、ADTSヘッダ自体は暗号化されていないので、各AOB_FRAMEにおけるADTSヘッダから、オーディオデータのデータ長を読み取ってゆけば、後続するAOB_FRAMEがどこから存在するかを知得することができる。以上でAOB_FRAMEについての説明を終える。

【 0 0 4 7 】

{3-3_10-4} AOB_ELEMENTについて

続いて図 1 0 において第 4 段目に位置するAOB_ELEMENTについて説明する。

『AOB_ELEMENT』は、連続する複数のAOB_FRAMEの集合である。ここで、どれだけの数のAOB_FRAMEがAOB_ELEMENTに含まれるかは、図 1 1 に示したsampling_frequency_indexの設定に従って変化する。即ち、AOB_ELEMENTに含まれるAOB_FRAMEの個数は、そのAOB_ELEMENTに含まれるAOB_FRAMEの再生時間が大体2秒になるように定められており、各サンプリング周波数に応じて、異なる個数となる。

【 0 0 4 8 】

{3-3_10-5_14} AOB_ELEMENTに含まれるAOB_FRAME数

図 1 4 は、sampling_frequencyと、AOB_ELEMENTに含まれるAOB_FRAME数との対応を示す図である。本図において、sampling_frequencyが48kHzである場合、AOB_ELEMENTに含まれるフレーム数は、94個となり、sampling_frequencyが44.1kHz

である場合、AOB_ELEMENTに含まれるフレーム数は87個、sampling_frequencyが32kHzである場合、AOB_ELEMENTに含まれるフレーム数は63個、sampling_frequencyが24kHzである場合、フレーム数は47個、sampling_frequencyが22.05kHzである場合、AOB_ELEMENTに含まれるフレーム数は44個、sampling_frequencyが16kHzである場合、AOB_ELEMENTに含まれるフレーム数は32個となる。但し、AOBを分割などの編集を行った場合、AOBの先頭と最後のAOB_ELEMENTのAOB_FRAME数は、図 1 4 の個数より少なくなる場合がある。

【 0 0 4 9 】

AOB_ELEMENTには、ヘッダ等の特別な情報は付与されていないが、その代わりにそのデータ長がタイムサーチテーブルに示されている。

{3-3_10-6_15} AOB_ELEMENT及びAOB_FRAMEの時間長の一例

図 1 5 は、AOB_ELEMENTの時間長及びAOB_FRAMEの時間長の一例を示す図である。本図の第 1 段目は、複数AOB_BLOCKの並びであり、第 2 段目は、複数AOB_ELEMENTの並びを示す。第 3 段目は、複数AOB_FRAMEの並びを示す。

【 0 0 5 0 】

本図を参照すると、AOB_ELEMENTは、約2.0秒という再生時間長に相当し、本図におけるAOB_FRAMEは、20msecという再生時間長に対応することが判る。AOB_ELEMENTのそれぞれに付されている『TMSRT_entry』という文字列は、各AOB_ELEMENTのデータ長がタイムサーチテーブルに記載されていることを示す。このようなTMSRT_entryを参照して、順方向サーチ再生、逆方向サーチ再生を行うことにより、例えば2.0秒をスキップして、240ミリ秒分だけ再生するという間欠な再生を実現することができるのである。

【 0 0 5 1 】

{3-3_10-7} AOB_BLOCKについて

以上でAOB_ELEMENTについての説明を終え、続いてAOB_ELEMENTの上位、即ち、図 1 0 のAOBファイルのデータ構成を示す図における第 3 段目に位置するのAOB_BLOCKについて説明する。

『AOB_BLOCK』は、有効なAOB_ELEMENTからなる領域であり、AOBファイル中に一つ存在する。AOB_ELEMENTが2秒という再生時間に相当するのに対して、AOB_BLOCK

OCKは8.4分の再生時間を上限とした再生時間に相当する。各AOBを8.4分の再生時間に限定した理由は、AOB_BLOCKに含まれるAOB_ELEMENTの個数を制限することにより、タイムサーチテーブルのサイズを504バイト以下に抑制するためである。

【0052】

{3-3_10-8} タイムサーチテーブルの抑制

以下、再生時間の限定により、タイムサーチテーブルの抑制が可能となった理由を詳細に説明する。

順方向サーチ再生、逆方向サーチ再生の再生を行う際、2秒分読み出しをスキップして240ミリ秒だけ再生するという『2秒スキップ240ミリ秒再生』が行われる。このように2秒という時間長をスキップする場合、原則として、AOB_FRAMEのADTSヘッダに示されているデータ長を順次参照してゆけばよいのだが、その場合、2秒という時間間隔をスキップするために100個(=2秒/20ミリ秒)ものAOB_FRAMEを順次検出せねばならず、再生装置に余分な処理負荷を与えてしまう。そのような処理負荷を軽減するには、その2秒間隔の読出先アドレスをタイムサーチテーブルに記述して、順方向サーチ再生及び逆方向サーチ再生が命じられた際、再生装置がこれを参照すればよい。即ち、タイムサーチテーブルには、2秒先、4秒先の読出先アドレスを算出するための情報、具体的には、各AOB_ELEMENTについてのデータ長を記述しておき、再生装置は、これを参照して、順方向サーチ再生・逆方向サーチ再生を行えばよいのである。2秒に相当するデータ長がどの程度になるかについて考察する。オーディオデータの再生時のビットレートは、上述したように16Kbps～144Kbpsの範囲であるので、2秒当たりに再生されるデータ長は4Kbyte(=16Kbps×2/8)～36Kbyte(=144Kbps×2/8)となる。

【0053】

2秒当たりのデータ長が4Kbyte～36Kbyteであるなら、オーディオデータのデータ長が記述されるためのタイムサーチテーブル内のエントリーのデータ長は、2バイト(16ビット)必要となる。何故なら、エントリーに16ビット長を割り当てたならば、0～64KByteの数値が記述されることができるからである。一方、タイムサーチテーブルの総データサイズを例えば504バイト(これは後述するTKTMSRTのデータサイズである)内に制限する場合を考えると、このタイムサーチテーブル

内に設けるべきエントリーは、 $252(=504/2)$ 個に制限せねばならない。上述したように、エントリーは、2秒毎に設けられるものであるので252エントリーに対応する再生時間は、504秒($=2\text{秒} \times 252$)となり、8分24秒($=8.4\text{分}$)となる。このようにAOB_BLOCKにおける再生時間を8.4分以下に制限したことにより、タイムサーチテーブルのデータサイズを504バイト以下とすることができる。

【 0 0 5 4 】

{3-3_10-9} AOBについて

以上でAOB_BLOCKについての説明を終え、続いてAOBについて説明する。

図 1 0 の第 2 段目に位置するAOBは、AOB_BLOCKの前後に無効領域が付与された領域であり、AOBファイル中に一つ存在する。

この無効領域は、当該、AOB_BLOCKと同じクラスタに格納され、当該AOB_BLOCKと共に読み書きされる領域である。AOBにおいて、何処から何処までがAOB_BLOCKに該当するのかは、ナビゲーションデータに含まれるBIT(その詳細についての説明は、後段で行う。)にて指定される。

【 0 0 5 5 】

以上で、各AOBファイルにどのようなデータが格納されているかが明らかとなった。続いて、図 9 に示した8つのAOBファイルに含まれるAOB、AOB_BLOCKが連続して読み出されることにより、どのような内容が再生されるかを説明する。

{3-3_10-10_16}

図 1 6 は、AOBファイルに収録されている各AOB、AOB_BLOCKが連続して再生されることにより、どのような再生内容が再生されるかを示す。第 1 段目は、ユーザ領域における8つのAOBファイルを示し、第 2 段目は、各AOBファイルに収録されている8つのAOBを示す。第 3 段目は、それぞれのAOBに含まれる8つのAOB_BLOCKを示す。

【 0 0 5 6 】

第 5 段目は、SongA、SongB、SongC、SongD、SongEという5つのコンテンツ部（この場合、これらは曲という単位を意味する）からなる音楽アルバムを示す。破線AS1,AS2,AS3……AS7,AS8は、音楽アルバムの分割部分と、AOB_BLOCKとの対応関係を示し、第 4 段目は、第 5 段目の音楽アルバムがどのような単位で分割され

るかを示す。

【0057】

これらの破線を参照すると、各AOB#1に含まれるAOB_Blockは、6.1分.という時間にて再生される曲(SongA)であり、各AOB#2に含まれるAOB_Blockは、3.3分.という時間にて再生される曲(SongB)、各AOB#3に含まれるAOB_Blockは、5.5分という時間にて再生される曲(SongC)である。以上のようにAOB001.SA1～AOB003.SA1は、それぞれが独立した曲に対応するものであることがわかる。

【0058】

一方、AOB#4は、30.6分という時間にて再生される曲(SongD)の先頭部分であり、8.4分という再生時間にて再生される。AOB#5、AOB#6に含まれるAOB_BLOCKはSongDの中間部分であり、8.4分という再生時間、AOB#7に含まれるAOB_BLOCKは、SongDの終端部分であり、5.4分という再生時間にて再生される。このように30.6分という再生時間を有する曲は、(8.4分+8.4分+8.4分+5.4分)という単位で分割され、各AOBに含まれていることがわかる。この図からも理解できるように、AOBファイルに含まれる全ての曲は、再生時間長が8.4分という時間長以内に収められていることがわかる。

【0059】

以上の説明によりAOBの再生時間長を制限することにより、各AOBに対応づけられているタイムサーチテーブルのデータサイズも制限されていることが明らかとなった。続いて、このタイムサーチテーブルを含むナビゲーションデータについて説明する。

{3-3_8(a)(b)-2}

ナビゲーションデータは、『SD_Audio.PLM』『SD_Audio.TKM』という2つのファイルからなることは既に述べた通りである。ファイル『SD_Audio.PLM』は、プレイリストマネージャ(Playlistmanager)を含み、ファイル『SD_Audio.TKM』は、トラックマネージャ(TrackManager)を含む。

【0060】

プレゼンテーションデータの説明で述べたように、複数のAOBファイルは、符号化されたAOBを収録しているが、これらのAOBの再生時間がどれだけであるか、

また、それぞれのA0Bがどのような曲名であり、作曲者は誰であるか等は何等記載されていない。このような情報、即ち、複数A0Bのそれぞれについての諸特性を再生時にこれらを再生装置に通知するべくトラックマネージャーは設けられている。一方、複数のA0Bは、複数のA0Bファイルに収録されているのみなので、それらをどのような順序で再生させるかは一切記載されていない。プレイリストマネージャーは、それらA0Bをどのような順序で再生させるべきかを再生装置に通知するべく設けられている。

【0061】

ここでトラックマネージャーは、A0Bファイルに収録されているA0Bを、トラックとして管理する複数のトラック管理情報を含む。トラックとは、ユーザにとって意味のある再生単位であり、フラッシュメモリカード31に音楽著作物を格納しようとする場合、トラックは曲に対応し、フラッシュメモリカード31にリーディングブックを格納しようとする場合（リーディングブックとは、書籍ではなく、読み上げ音声により表現された文書著作物をいう）、ブックジャンルであるなら、トラックは、文の章／節に対応する。トラックマネージャーは、複数A0Bファイルに収録されている複数のA0Bをトラックの集合として管理するために設けられている。

【0062】

プレイリストとは、トラックの複数の再生順序を規定するものであり、プレイリストマネージャーは、このようなプレイリストを複数含んでいる。

以降、トラックマネージャーについて図面を参照しながら説明する。

{17-1_18} Playlistmanager及びTrackManagerの詳細構成

図17は、実施形態におけるPlaylistmanager及びTrackManagerの構成を段階的に詳細化した図であり、図18は、PlayListManager及びTrackManagerのサイズを示す図である。即ち、本図において右段に位置する論理フォーマットは、その左段に位置する論理フォーマットを詳細化したものであり、破線に示す引き出し線は、右段の論理フォーマットがその左段の論理フォーマット内のどの部分を詳細化したかを明確にしている。

【0063】

このような表記に従って図 1 7 におけるTrackManagerの構成を参照すると、TrackManagerは、破線の引き出し線h1に示すように、Track Information(TKIと略す)#1,#2,#3,#4……#nからなる。これらのTKIはAOBファイルに収録されているAOBを、トラックとして管理するための情報であり、各AOBファイルに対応している。

【 0 0 6 4 】

図 1 7 を参照すると各TKIは、破線の引き出し線h2に示すように、Track_General_Informatin(TKGI)、TKIに固有なテキスト情報が記述されるTrack_Text_Information(TKXTI_DA)、タイムサーチテーブルの役割を有するTrack Time Serch Table(TKTMSRT)からなることがわかる。図 1 8 を参照すると、TKI自体は固定サイズ(1024バイト)であり、TKGIとTKXTI_DAとは合計で512バイト固定長であることがわかる。TKTMSRTも512バイト固定長である。またTrackManagerにおいて、TKIは、最大999個まで設定することができる。

【 0 0 6 5 】

このTKTMSRTは、破線の引き出し線h3に示すように、TMSRT_Headerと、TMSRT_entry#1,#2,#3……#nとからなることがわかる。

{17-2_19} TKIと、AOBファイル及びAOBとの相互関係

図 1 9 は、図 1 7 に示したTKIと、図 1 6 に示したAOBファイル及びAOBとの相互関係を示す図である。図 1 9 の第 1 段目における四角枠はTrackManagerを示し、第 2、第 3、第 4 段目は図 1 6 に示した8つのAOBファイルを示す。第 4 段目における8つの枠は、8つのAOBを示す。この8つのAOBファイルは、図 1 6 に示した8つのAOBを収録していたものであり、SongA、SongB、SongCを含む音楽アルバムを形成している。第 1 段目は、8つのTKIを示す。これらTKIに付与された数値”1”, ”2”, ”3”, ”4”は、各TKIを識別するためのシリアル番号であり、各TKIは、同じシリアル番号001,002,003,004,005……が付与されたAOBファイルと対応づけられている。この点に注意して、図 1 9 を参照すれば、TKI#1がAOB001.SA1に対応していて、TKI#2がAOB002.SA1、TKI#3がAOB003.SA1、TKI#4がAOB004.SA1に対応していることがわかる(本図における矢印TA1,TA2,TA3,TA4……は、各TKIがどのAOB_FRAMEと対応しているかを示す。)。このように各TKIは、各AOBファイルに収

録されているAOBと、1対1の対応関係を有するので、各TKIには、AOBに固有な情報を詳細に記載しておくことができる。

【 0 0 6 6 】

{17-3_20} TKTMSRTのデータ構造について

AOBファイルに収録されているAOBに固有な情報として、先ず初めにTKTMSRTについて説明する。図 2 0 は、図 1 7 に示したTKTMSRTの詳細なデータ構造を示す図である。本図の右側には、タイムサーチテーブルヘッダ(TMSRT_Header)の詳細なデータ構造が示されている。図 2 0 において、タイムサーチテーブルヘッダのデータサイズは8バイトであり、TMSRT_ID(0バイト目から1バイト目まで)、reserved(2バイト目から3バイト目まで)、Total TMSRT_entry_Number(4バイト目から7バイト目まで)という3つのフィールドを有する。『TMSRT_ID』には、TMSRTを一意に識別できるIDが記述される。『Total TMSRT_entry Number』には、当該TMSRT内にあるTMSRT_entryの総数が記述される。

【 0 0 6 7 】

{17-3_21-1} TKTMSRTの具体例について

続いてTKTMSRTについてより詳細に説明する。図 2 1 は、TKTMSRTについての一例を示す図である。本図の左側に、AOBを示し、右側にTKTMSRTを示す。本図左側のAOBは、複数のAOB_ELEMENT#1,#2,#3……#n-1からなり、その右側における複数の領域AR1,AR2,AR3……ARnを占有している。また図中の『0』『32000』『64200』『97000』『1203400』『1240000』といった数値は、AOBに含まれるAOB_BLOCK先頭からの各AOB_ELEMENTの占有領域AR1,AR2,AR3,ARn-1,ARnまでの相対アドレスを示す。AOB_ELEMENT#2は、AOB_BLOCK先頭から『32000』だけ隔てられた位置に記録されていることを示す。AOB_ELEMENT#3は、AOB_BLOCK先頭から『64200』だけ隔てられた位置に、AOB_ELEMENT#n-1は、AOB_BLOCK先頭から『1203400』だけ隔てられた位置に記録されていることを示す。

【 0 0 6 8 】

注意すべきは、各占有領域の先頭アドレスの間隔が一定値ではないこと、即ち、各AOB_ELEMENTの占有領域が、それぞれ異なるサイズだけ複数クラスタを占有していることである。各占有領域のサイズがそれぞれ異なるのは、各AOB_FRAME

における符号割り当てが可変長だからである。

各AOB_ELEMENTの占有サイズが異なるので、各AOB_ELEMENTの先頭にジャンプする場合、各AOB_ELEMENTがAOB内の何処に存在するかを予め再生装置に指示しておく必要がある。このような目的をもって、複数のTMSRT_entryは記載されている。矢印RT1,RT2,RT3……RTn-1,RTnは、これら各AOB_ELEMENTの占有領域AR1,AR2,AR3……ARn-1,ARnと、TMSRT_entry#1、TMSRT_entry#2、TMSRT_entry#3……TMSRT_entry#n-1,TMSRT_entry#nとの対応関係を示す。即ち、AOB_ELEMENT#1の占有領域AR1がどれだけのサイズを占有しているかがTMSRT_entry#1に記載され、AOB_ELEMENT#2、AOB_ELEMENT#3の占有領域AR2,AR3がどれだけのサイズを占有しているかがTMSRT_entry#2、TMSRT_entry#3に記載される。

【 0 0 6 9 】

ここで、占有領域AR1は、AOBの先頭から、AOB_ELEMENT#2の先頭『32000』迄を占有しているので、TMSRT_entry#1は32000(=32000-0)と記述され、占有領域AR2は、AOB_ELEMENT#1の先頭『32000』から、AOB_ELEMENT#2の先頭『64200』迄を占有しているので、TMSRT_entry#2は『32200(=64200-32000)』と記述、占有領域AR3は、AOB_ELEMENT#3の先頭『64200』から、AOB_ELEMENT#4の先頭『97000』迄を占有しているので、TMSRT_entry#3は『32800(=97000-64200)』、占有領域ARn-1は、AOB_ELEMENT#n-1の先頭『1203400』から、AOB_ELEMENT#nの先頭『1240000』迄を占有しているので、TMSRT_entry#n-1は『36600(=1240000-1203400)』と記述されている。

【 0 0 7 0 】

{17-3_21-2} TKTMSRTの読み出し方式

このようにタイムサーチテーブルには、AOB_ELEMENTのデータサイズが記載されていることがわかる。一方、AOB_ELEMENTの説明で述べたように、各AOB_BLOCKのデータ長は、再生時間が8.4分内になるように定められているので、1つのAOBに含まれるAOB_ELEMENTの総数は、所定数(図 2 0 に示す252個)以下に抑えられている。AOB_ELEMENT数が所定数以下に抑えられるので、AOB_ELEMENTに対応するTMSRT_entryの総数も所定数以下となり、これらを含むTKTMSRTのデータサイズも所定サイズ以下となる。TKTMSRTのサイズを抑制したため、再生装置は、以下のよう

にTKIを読み出して、利用することができる。

【0071】

あるAOBが読み出されて、その再生が開始されると、それに対応するTKIを読み出して、メモリに格納する。以降、当該AOBの再生が継続している期間において、このTKIをメモリに格納しておく。当該AOBの再生が終われば、これに後続するAOBが読み出されて、その再生が開始されると、それに対応するTKIを読み出して、それまでメモリ上に格納されていたTKIを、新たに読み出されたTKIを用いて上書きする。以降、当該AOBの再生が継続している期間において、このTKIをメモリに格納しておく。

【0072】

TKIの読み出しと、メモリへの格納とをこのように行えば、再生装置におけるメモリの実装量が小規模であっても、必要なTKIを読み出すだけで順方向サーチ再生、逆方向サーチ再生といった特殊再生を行うことができる。尚、本実施形態では、あるAOB_ELEMENTの先頭アドレスから次のAOB_ELEMENTの先頭アドレスまでのデータ長をTMSRT_entryとして記載したが、AOB_BLOCKの先頭から、各AOB_ELEMENTの先頭までの相対アドレスを記載してもよい。

【0073】

{17-3_21-3} AOB_ELEMENTを含むクラスタの特定

最後にTKTMSRTを参照して、任意のAOB_ELEMENTをどうやって読み出せばよいかについて説明する。各AOB_ELEMENTのサイズが記載されたTKTMSRTを参照して、AOBにおいて先頭からy番目に位置するAOB_ELEMENT#yを読み出す場合、以下の{数式1}を満たすクラスタuを求めて、そのクラスタuの先頭からオフセットv以降を読み出せばよい。

{数式1}

クラスタu = (AOB_ELEMENT#1からAOB_ELEMENT#y-1までのTMSRT_entryの総和 + DATA_Offset) / クラスタサイズ

オフセットv = (AOB_ELEMENT#1からAOB_ELEMENT#y-1までのTMSRT_entryの総和 + DATA_Offset) mod クラスタサイズ

※ $c = a \bmod b$ とある場合、cは、aをbで割った場合の余りを示し、DATA_Off

setは、BITに記載されている情報であり、後述する。

【 0 0 7 4 】

{17-4} TKTXI_DAについて

以上で、タイムサーチマップ(TKTMSRT)の説明を終わる。次に、図 1 7 においてTKTMSRTの上段に記載されているTrack Text Information Data Area(TKTXI_DA)について説明する。

Track Text Information Data Area(TKTXI_DA)には、アーティスト名、アルバム名、編曲者名、プロデューサ名等を示すテキスト情報が記述される。テキストデータが存在しない場合でも、この領域は確保される。

【 0 0 7 5 】

{17-5} TKGIについて

続いてTKTXI_DAの上段にあるTKGIについて説明する。図 1 7 においてTKIのTKGIは、破線の引き出し線h4に示すように、TKIの識別子『TKI_ID』、TKI番号『TKI_IN』、TKIのサイズ『TKI_SZ』、次のTKIへのリンクポインタ『TKI_LNK_PTR』、ブロック属性『TKI_BLK_ATR』、再生時間『TKI_PB_TM』、TKIのオーディオ属性『TKI_AOB_ATR』、『ISRC』、ブロック情報『BIT』という一連の情報が記録されていることがわかる（尚、本図は、説明の簡略化のため、一部のフィールドについては省略して表記している。）。

【 0 0 7 6 】

{17-5_22-1} TKGIについて

以下、図 2 2 を参照しながらTKGIの詳細構成について説明する。本図と、図 2 2 との違いは、図 1 7 に示したTKGIのデータ構成が図中左側に配置されており、図 1 7 では明らかにされてなかった『TKI_BLK_ATR』、『TKI_AOB_ATR』、『ISRC』のビット構成が、図中の右側に配置されている点である。

【 0 0 7 7 】

{17-5_22-2} TKI_IDについて

『TKI_ID』には、TKIを一意に識別できるID(本実施形態では2バイトの"A4"というコード)が記述される。

{17-5_22-3} TKINについて

『TKIN』には、1から999までの範囲のTKI番号が記述される。なお、このTKI番号は他のTKIのTKINに記述されるTKI番号と重複してはならない。このようなTKINとして、TrackManagerにおけるTKIの順位、即ち、TrackManagerにおいてTKIが何番目に位置するかを記述するものとする。本図におけるTKI#1なら、TKI番号は、"1"と記載され、TKI#2ならTKI番号は、"2"と、TKI#3ならTKI番号は、"3"と記載される。

【0078】

{17-5_22-4} TKI_SZについて

『TKI_SZ』には、TKIのデータサイズがバイト数単位で記述される。図22では、TKIのデータサイズが1024バイトと規定されているので、本実施形態において1024バイトと記述される。

{17-5_22-5} TKI_LNK_PTRについて

『TKI_LNK_PTR』には、当該TKIのリンク先のTKIについてのTKINが記述される。ここで、TKI間の対応関係について説明する。

【0079】

曲が複数のAOBから構成され、それらが複数のAOBファイルに収録されている場合、それら複数のAOBファイルに対応づけられている複数のTKIは一体となって、当該曲を管理することになる。このように複数のTKIが一体となっている場合、これらTKIに対応するAOBファイルに、どのTKIに対応するAOBファイルが後続するかを示す必要がある。TKI_LNK_PTRは、各TKIに後続するTKIについてのTKINを記述するという用途に用いられる。

【0080】

{17-5_22-6_19} TKI_LNK_PTRについて

以降、図19に示した8つのTKIにおいて、TKI_LNK_PTRがどのように設定されているかについて説明する。1曲を構成するTKI#1～TKI#3、TKI#8において、そのTKI_LNK_PTRは設定されないが、曲SongDを構成する4つのAOBファイルに対応するTKI#4、TKI#5、TKI#6、TKI#7は、各TKI_LNK_PTRが次のTKI_LNK_PTRを指示するよう設定されている。即ち、矢印TL4, TL5, TL6に示すように、TKI#4のTKI_LNK_PTRはTKI#5を指示しており、TKI#5のTKI_LNK_PTRはTKI#6を、TKI#6のTKI_LNK_PTR

はTKI#7を指示している。これらは、何れもSongDを構成する。4つのAOBファイルに対応づけられているTKIにおけるこれらTKI_LNK_PTRを参照することにより、TKI#4～TKI#7という4つのTKI、及びAOB004.SA1～AOB007.SA1という4つのAOBファイルが、一体となってSongDを構成しているということがわかる。

【0081】

{17-5_22-7} TKI_BLK_ATRについて

『TKI_BLK_ATR』には、TKIについての属性が記述される。図22においてTKI_BLK_ATRから破線にて引き出された枠に、TKI_BLK_ATRのビット構成を示す。本図においてTKI_BLK_ATRは16ビットであり、b3ビットからb15ビットまでが将来の拡張のために確保されている。ビット番号b2からb0までの3ビットを用いて、TKIについての属性が記述される。

【0082】

TKIが使用されており、1個のTKIの中に1個の曲が入っている場合、TKI_BLK_ATRには"000b"の値が記述される(以降、この設定を『Song』という。)。TKIが使用されており、1曲が複数のTKIから構成され、当該TKIがその先頭である場合は、TKI_BLK_ATRには"001b"の値が記述される(以降、この設定を『Head_of_Song』という。)。TKIが使用されており、1曲が複数のTKIから構成され、当該TKIがその中間である場合は、TKI_BLK_ATRには"010b"の値が記述される(以降、この設定を『Midpoint_of_Song』という)。TKIが使用されており、1曲が複数のTKIから構成され、当該TKIがその終端である場合、TKI_BLK_ATRには"011b"の値が記述される(以降、この設定を『End_of_Song』という。)。TKIが未使用であり、TKIの領域がある場合、すなわち削除されたTKIである場合は、"100b"の値が記述される(以降、この設定を『Unused』という)。TKIが未使用であり、TKIの領域がない場合、すなわち初期状態のTKIである場合は、"101b"の値が記述される。

【0083】

{17-5_22-8_19} TKI_BLK_ATRの設定例

図19の一例では、それぞれのTKIについてのTKI_BLK_ATRがどのように設定されているかについて説明する。

各TKIにおけるTKI_BLK_ATRを参照すれば、TKI#1 (AOB001.SA1)、TKI#2 (AOB002.SA1)、TKI#3 (AOB003.SA1)、TKI#8 (AOB008.SA1) という4つの組みは、それぞれが独立した曲に対応しているので、TKI#1、TKI#2、TKI#3、TKI#8のTKI_BLK_ATRは、『Song』と設定されている。

【0084】

TKI#4におけるTKI_BLK_ATRは『Head_of_Song』と設定され、TKI#7におけるTKI_BLK_ATRは『End_of_Song』と、TKI#5、TKI#6は『Midpoint_of_Song』と設定されていることがわかる。このことは、TKI#4と対応関係を有するTKI#4 (AOB004.SA1) は曲の先頭部、TKI#5、TKI#6と対応関係を有するTKI#5 (AOB005.SA1) 及びTKI#6 (AOB006.SA1) は曲の中間部、TKI#7と対応関係を有するTKI#7 (AOB007.SA1) は曲の終端部であることを意味する。

【0085】

このように各TKIにおけるTKI_BLK_ATRの記載に従って、TKI (AOBファイル) の組みを分類すれば、TKI#1 (AOB001.SA1) の組みが1つ目の曲(SongA)を構成していることがわかる。TKI#2 (AOB002.SA1) の組みが2つ目の曲(SongB)、TKI#3 (AOB003.SA1) の組みが3つ目の曲(SongC)を構成していることがわかる。

TKI#4 (AOB004.SA1) の組みが4つ目の曲(SongD)の先頭部分を構成しており、TKI#5 (AOB005.SA1) の組みと、TKI#6 (AOB006.SA1) の組みとがSongDの中間部分を構成しており、TKI#7 (AOB007.SA1) の組みとがSongDの終端部分を構成していることがわかる。TKI#8 (AOB008.SA1) の組みとは独立して5つ目のSongEの終端部分を構成していることがわかる。

【0086】

{17-5_22-9} TKI_PB_TMについて

『TKI_PB_TM』には、TKIに対応するAOBファイルに収録されているAOBにより構成されるトラック(曲)の再生時間が記述される。

トラック(曲)が1つのAOBファイルから構成される場合、『TKI_PB_TM』にはそのAOBファイルにおける再生時間が記述される。上述したように、それぞれのAOBは、再生時間が一律に8.4分以下になるように分割されているので、トラック(曲)が1つのAOBファイルから構成される場合は、そのトラックに対応するTKIにおい

てこのTKI_PB_TMは、8.4分以下の再生時間が記述される。

【 0 0 8 7 】

トラック(曲)が複数のAOBファイルから構成される場合、『TKI_PB_TM』にはそれらAOBファイルにおける再生時間の合計時間が記述される。この場合、『TKI_PB_TM』には、8.4分以上の再生時間が記述されることになる。

{17-5_22-10} TKI_AOB_ATRについて

『TKI_AOB_ATR』には、TKIに対応するAOBファイルに収録されているAOBがどのようなサンプリング周波数でサンプリングされているか、どのようなビットレートで転送されるか、チャンネル数がどれだけであるか等、AOBを生成する際のエンコード条件が記述される。『TKI_AOB_ATR』から破線にて引き出された枠は、TKI_AOB_ATRのビット構成を示す。本図においてTKI_AOB_ATRは、16ビットであり、ビット番号b13からビット番号b11までのフィールドには、コーディングモードが記述される。MPEG-2 AAC(with ADTS header)でエンコードされている場合には、“000b”の値が記述される。

【 0 0 8 8 】

ビット番号b10からビット番号b8までのフィールドには、ビットレートが記述される。64kbpsの場合は、“000b”の値が記述される。32kbpsの場合は“001b”、16kbpsの場合は“010b”の値が記述され、ビット番号b7からビット番号b4には、サンプリング周波数が記述される。48kHzの場合は“0000b”、44.1kHzの場合は“0001b”、32kHzの場合は“0010b”の値が記述される。

【 0 0 8 9 】

ビット番号b3からビット番号b1までのフィールドには、チャンネル数が記述される。1ch(mono)の場合は、“000b”が記述される。2ch(stereo)の場合は、“001b”が記述される。

ビット番号b15からビット番号b14、およびビット番号b0の領域は、将来の拡張用に予約されている。

【 0 0 9 0 】

{17-5_22-11} ISRCについて

『ISRC』には、TKGIにおけるISRC (International Standard Recording Code

）が記述される。図 2 2 における『ISRC』から破線にて引き出された枠はISRCの内容を示す。この枠に示されているように、ISRCは、10バイトからなり、ビット番号b4からビット番号b7までのフィールドにRecrding-item code(#12)が記述され、ビット番号b8からビット番号b11までのフィールドにRecrding code/Recrding-item code(#11)が記述される。

【 0 0 9 1 】

ビット番号b12からビット番号b23までのフィールドにRecrding code(ISRC#10, #9, #8)が記述される。ビット番号b24からビット番号b31までのフィールドにYear-of-Recrding code(ISRC#6, #7)が記述される。

以降、ビット番号b32からビット番号b37までのフィールド、ビット番号b40からビット番号b45までのフィールド、ビット番号b48からビット番号b53までのフィールド、ビット番号b56からビット番号b61までのフィールド、ビット番号b64からビット番号b69までのフィールドには、Country code(ISRC#1, #2, #3, #4, #5)が記述される。ビット番号b79のフィールドには、1ビットのValidity flagが記述される。尚、ISRCの詳細については、IS03901 : 1986 ' ' Documentation-International Standard Recording Code (ISRC) ' ' を参照されたい。

【 0 0 9 2 】

{17-5_22-12_23(a)-1} BITについて

『ブロック情報テーブル(BIT)』は、A0B_BLOCKを管理するテーブルである。図 2 3 (a) (b) は、BITの詳細構成を示す図である。図 2 3 (a) に示すように、BITは、60バイト目から63バイト目までを占めるDATA_OFFSETフィールドと、64バイト目から67バイト目までを占めるSZ_DATAフィールドと、68バイト目から71バイト目までを占めるTMSRTE_Nsフィールドと、72バイト目から73バイト目までを占めるFNs_1st_TMSRTEフィールドと、74バイト目から75バイト目までを占めるFNs_Last_TMSRTEフィールドと、76バイト目から77バイト目までを占めるFNs_Middle_TMSRTEフィールドとからなる。以下、各構成要素の説明を行う。

【 0 0 9 3 】

{17-5_22-12_23(a)-2} DATA_Offsetについて

『DATA_OFFSET』には、クラスタ境界から各A0B_BLOCKの先頭までの相対アドレ

スがバイト単位で記述される。これにより、AOBからAOB_BLOCKまでの間に無効領域がどれだけ存在するかが表現される。AOBとしてフラッシュメモリカード 3 1 に格納されている音楽が、エアチェックして録音された音楽であり、その音楽のイントロの部分にディスクジョッキーの音声が混じっている場合、BITにおけるDATA_Offsetを設定することにより、この不要音声をAOB_BLOCKから除外して再生させないようにすることができる。

【 0 0 9 4 】

{17-5_22-12_23(a)-3} SZ_DATAについて

『SZ_DATA』には、各AOB_BLOCKのデータ長がバイト単位で記述される。SZ_DATAとDATA_Offsetとを加算した値をAOBを収録しているファイルサイズ(クラスタサイズの整数倍)から差し引けば、AOB_BLOCKに後続する無効領域がどれだけのサイズであるかを求めることができる。

【 0 0 9 5 】

{17-5_22-12_23(a)-4} TMSRTE_Nsについて

『TMSRTE_Ns』には、各AOB_BLOCKに含まれるTMSRT_entryの総数が記述される。

{17-5_22-12_23(a)-5} 『Fns_1st_TMSRTE』、 『Fns_Last_TMSRTE』、 『Fns_Middle_TMSRTE』について

『Fns_1st_TMSRTE』には、当該AOB_BLOCK中の先頭に位置するAOB_ELEMENTに含まれるAOB_FRAME数が記述される。

【 0 0 9 6 】

『Fns_Last_TMSRTE』には、AOB_BLOCKの最後尾のAOB_ELEMENTに含まれるAOB_FRAMEの個数が記述される。

『Fns_Middle_TMRTE』には、先頭と最後尾のAOB_ELEMENTを除くAOB_ELEMENT、即ち、AOB_BLOCKの中間部に位置するAOB_ELEMENTに含まれるAOB_FRAMEの個数が記述される。

【 0 0 9 7 】

{17-5_22-13_23(b)}

図 2 3 (b) は、Fns_Middle_TMRTEにAOB_FRAMEが幾つ格納されているかを示

す図である。本図は図 1 4 同様、sampling_frequencyと、中間部のAOB_ELEMENTに含まれるAOB_FRAME数との対応関係を示している。本図におけるsampling_frequencyと、AOB_ELEMENTに含まれるフレーム個数との対応関係は図 1 4 と全く同一であり、サンプリング周波数に応じて異なる個数になっていることがわかる。『FNS_1st_TMSRTE』及び『FNS_Last_TMSRTE』におけるフレーム数は、『FNS_Middle_TMSRTE』におけるフレーム数と原則同一のフレーム数に設定されるが、AOB_BLOCKの先頭又は末尾に位置するAOB_ELEMENTに無効領域を設定する場合、『FNS_1st_TMSRTE』及び『FNS_Last_TMSRTE』は、『FNS_Middle_TMSRTE』と異なる値となる。

【 0 0 9 8 】

{17-5_22-14_24} AOB_ELEMENTの格納例

図 2 4 は、AOB_ELEMENT#1～#4からなるAOBが格納されているクラスタ007～クラスタ00Eを示す図である。AOBが図 2 4 に示すように格納されている場合に、BITがどのように設定されるかについて説明する。これらクラスタ007～クラスタ00Eに格納されているAOB_ELEMENT#1～AOB_ELEMENT#4のそれぞれには、三角旗状の記号が付与されているが、これらは、AOB_ELEMENT#1～AOB_ELEMENT#4のそれぞれに、TKIに含まれるTMSRT_entryが設定されていることを示す。

【 0 0 9 9 】

この際、AOB先端におけるAOB_ELEMENT#1は、クラスタ007に格納されており、AOB末尾におけるAOB_ELEMENT#4は、クラスタ00Eに格納されている。AOB_ELEMENT#1～#4は、クラスタ007の途中md0からクラスタ00Eの途中md4迄を占有している。BIT内のSZ_DATAは、矢印sd1に示すようにAOB_ELEMENT#1からAOB_ELEMENT#4の最後までを指示しており、クラスタ007,00E内の領域であって、AOB_ELEMENTにより占有されていない部分ud0,ud1を指示していない。

【 0 1 0 0 】

これに対して、AOBは、クラスタ007、クラスタ00E内の領域であって、AOB_ELEMENT#1、AOB_ELEMENT#4により占有されていない部分ud0,ud1までも含んでいる。BIT内のDATA_Offsetは、非占有部分ud0のデータ長、即ち、クラスタ007の先頭から、AOB_ELEMENT#1の先頭までの相対値を指示している。

本図においてAOB_ELEMENT#1は、クラスタ007の途中md0からクラスタ008の途中md1までを占有している。このAOB_ELEMENT#1は、クラスタ008全体を占有しているのではなく、その終端部分以降は、AOB_ELEMENT#2に占有されている。AOB_ELEMENT#4は、クラスタ00Cの途中部分md3から、クラスタ00Eの途中部分md4までを占有している。このようにAOB_ELEMENTには、クラスタの境界を跨ぐように、記録されているものが存在することがわかる。つまり、AOB_ELEMENTは、クラスタの境界とは全く関係無く、記録されているのである。BIT内の『FNS_1st_TMSRTE』は、クラスタ007～クラスタ008におけるAOB_ELEMENT#1のフレーム数を示しており、BIT内の『FNS_Last_TMSRTE』は、クラスタ00C～クラスタ00EにおけるAOB_ELEMENT#4のフレーム数を示している。

【0 1 0 1】

このように、各AOB_ELEMENTは、クラスタの境界に関係なく、自由に配置されており、BITにより、クラスタ境界からAOB_ELEMENTまでのオフセットや各AOB_ELEMENT毎のフレーム数が管理されていることがわかる。

{17-5_22-14_25} 各AOB_ELEMENT毎のフレーム数の利用法 1

BITに記載されている各AOB_ELEMENT毎のフレーム数がどのように利用されるかを以下に説明する。BITに記載されているフレーム数は、先ず第1に、再生経過時刻を2秒スキップして、240ミリ秒だけ再生するという順方向サーチ再生、逆方向サーチ再生を行う場合に用いられる。

【0 1 0 2】

図25は、AOB内の任意のAOB_ELEMENT#yにおけるAOB_FRAME#xから順方向サーチ再生を行う場合、次に再生すべきAOB_FRAME#x+1をどのように設定するかを示す図である。

本図は、AOB_ELEMENT#yに含まれるAOB_FRAME#xが再生されている時点において、順方向サーチ再生が指示された場合を想定して作図した図である。本図において、 t は、所定の間欠再生時間(=240ミリ秒)、 $f(t)$ は、間欠再生時間に相当するフレーム数、間欠スキップ時間 $skip_time$ は、間欠再生を行う際にスキップすべき時間長(この場合は2秒)、この間欠スキップ時間 $skip_time$ に対応するフレーム数を $f(skip_time)$ とする。ここで間欠再生は、以下の①②③の手順を繰り返す

ことにより行われる。

【 0 1 0 3 】

①TKTMSRTに記載されているTMSRT_entryを参照して、旗(AOB_ELEMENT)の先頭へとジャンプする。

②240ミリ秒だけ再生を行う

③次の旗(AOB_ELEMENT)の先頭へとジャンプする。

AOB_ELEMENT#yに含まれるAOB_FRAME#xから、2秒+240ミリ秒後のAOB_FRAME#x+1は、AOB_ELEMENT#y+1内に存在する筈である。2秒+240ミリ秒後のAOB_FRAME#x+1を特定する場合、次のAOB_ELEMENT#y+1についての先頭アドレスは、TKTMSRTにおけるTMSRT_entryを読み出すことにより即座に算出することができるが、そのAOB_ELEMENT#y+1の先頭アドレスからAOB_FRAME#x+1までに介在するAOB_FRAME数は、TMSRT_entryのみでは知り得ない。そのようなAOB_FRAME数を算出するためには、AOB_FRAME#xがAOB_ELEMENT#yの先頭から何番目に位置するかを示す#xと、 $f(t)$ と、 $f(\text{skip_time})$ との和から、AOB_ELEMENT#yに含まれる全フレーム数を差し引くことにより求める必要がある。そのように、次のAOB_ELEMENT#y+1におけるAOB_FRAME#x+1の相対フレーム位置を簡易に算出するため、BITに各AOB_ELEMENTについての『Fns_1st_TMSRTE』、『Fns_Middle_TMSRTE』、『Fns_Last_TMSRTE』を記載しているのである。

【 0 1 0 4 】

{17-5_22-15_26(a)} 各AOB_ELEMENT毎のフレーム数の利用法2

BITに記載されているフレーム数は、第2に、任意の再生時刻から再生を開始するという機能（タイムサーチ機能）を実行する際に利用される。図26(a)は、任意の再生開始時刻が指定された場合、その指定時刻に対応するAOB_ELEMENT、AOB_FRAMEをどのように特定するかを示す図である。

【 0 1 0 5 】

任意の時刻が指定されて再生が指示された場合、再生指定時刻をJump_Entry(秒)とすると、以下の式を満たすAOB_ELEMENT#yと、AOB_FRAME位置xとから、再生を開始すればよい。

{数式 2}

$$\text{Jump_Entry(秒)} = (\text{Fns_1st_TMSRTE} + \text{Fns_middle_TMSRTE} \cdot y + x) \times 20\text{msec}$$

これら『Fns_1st_TMSRTE』及び『Fns_Middle_TMSRTE』はBITに記載されているので、これらを{数式 2}に適用することによりAOB_ELEMENT#y、AOB_FRAME#xが算出されれば、このAOBに対応するTKTMSRTを参照して、AOBにおいてy+2番目に位置するAOB_ELEMENT#y+2の先頭アドレスを求めて、この先頭アドレスから、AOB_FRAME#xの探索を始め、x番目のAOB_FRAMEが探索されれば、このx番目のAOB_FRAMEから再生を開始する。これにより、Jump_Entry(秒)にて指定された時刻から、再生を開始することができる。

【0 1 0 6】

この際、AOBファイルのADTSヘッダ部分を検索せず、TKTMSRTにTMSRT_entryが記述されているAOB_ELEMENT単位で検索を行えばよいので、再生指定時刻に対応する再生位置を高速に探し出すことができる。

同様に、複数のAOBからなる曲に対して、タイムサーチ機能が実行され、Jump_Entry(秒)が指定された場合、以下の{数式 3}を満たすAOB_ELEMENT#yと、AOB_FRAME#xとを算出すればよい。

{数式 3}

$$\begin{aligned} \text{Jump_Entry(秒)} = & \text{AOB\#1からAOB\#nまでの再生時間の総和} \\ & + (\text{Fns_1st_TMSRTE}(\#n+1) + \text{Fns_middle_TMSRTE}(\#n+1) \cdot y + x) \cdot 20\text{msec} \end{aligned}$$

ここでAOB#1からAOB#nまでのAOBの再生時間の総和は、以下の通りである。

AOB#1からAOB#nまでの再生時間の総和＝

$$\begin{aligned} & (\text{『Fns_1st_TMSRTE』}(\#1) + \text{『Fns_Middle_TMSRTE』}(\#1) \cdot \text{TMSRT_entry数}(\#1) + \text{『Fns_Last_TMSRTE』}(\#1) \\ & + \text{『Fns_1st_TMSRTE』}(\#2) + \text{『Fns_Middle_TMSRTE』}(\#2) \cdot \text{TMSRT_entry数}(\#2) + \text{『Fns_Last_TMSRTE』}(\#2) \\ & + \text{『Fns_1st_TMSRTE』}(\#3) + \text{『Fns_Middle_TMSRTE』}(\#3) \cdot \text{TMSRT_entry数}(\#3) + \text{『Fns_Last_TMSRTE』}(\#3) \end{aligned}$$

.....

+ 『Fns_1st_TMSRTE』(#n)+『Fns_Middle_TMSRTE』(#n)・TMSRT_entry数(#n)+『Fns_Last_TMSRTE』(#n))・20msec

{数式 3} を満たす AOB#n、AOB_ELEMENT#y、AOB_FRAME#x が算出されれば、この AOB#n+1 に対応する TKTMSRT を参照して、y+2 番目の AOB_ELEMENT#y+2 に位置するアドレスから、AOB_FRAME#x の探索を始め、x 番目の AOB_FRAME が探索されれば、この x 番目の AOB_FRAME から再生を開始する。

【0 1 0 7】

{17-5_22-16_27(a)(b)} AOBファイル及びTKIの削除

TKIに含まれる情報を全て説明したところで、一部の曲が削除された場合(case 1)、一部の曲が削除された後、新たな曲を記録する場合(case2)、複数の曲のうち、任意の2つを1つの曲に統合する場合(case3)、1つの曲を分割して、2つの曲を得る場合(case4)において、TKIがどのように更新されるかについて説明する。

【0 1 0 8】

先ず初めに、一部の曲が削除された場合(case1)について説明する。

図 2 7 (a) (b) は、曲を削除する場合を想定した図である。本図は、図 1 9 に示した TrackManager を示すものであり、本図において SongB を削除することを操作者が希望しているものとする。この SongB に対応する AOB は、AOB002.SA1 に収録されており、それが TKI#2 に対応づけられているので、AOB002.SA1 が削除されると共に、TKI#2 の TKI_BLK_ATR が『Unused』に設定される。AOB002.SA1 が削除され、TKI#2 の TKI_BLK_ATR が『Unused』に設定された状態を図 2 7 (b) に示す。AOB002.SA1 が削除されたので、データ領域において AOB002.SA1 が占有していた領域は空き領域に解放される。それと共に、TrackManager においては、TKI#2 の TKI_BLK_ATR が『Unused』に設定されていることがわかる。

【0 1 0 9】

{17-5_22-17_28(a)(b)} 新たに AOB ファイルを記録する場合の TKI の割り当て

続いて一部の曲が削除された後、新たな曲を記録する場合(case2)について説明する。

図 2 8 (a) は、曲の削除が複数回行われた後のTrackManagerを示す図である。本図において、複数の曲が削除され、これらがTKI#2、TKI#4、TKI#7、TKI#8に対応づけられているとすれば、これらのTKIのTKI_BLK_ATRが『Unused』に設定される。AOBファイルの削除は、通常のファイルと同様に行われるが、TrackManagerは、該当するTKIのTKI_BLK_ATRが『Unused』に設定されるのみで削除処理は完了する。そうすると、本図に示すように『Unused』のTKIが虫食い状にTrackManager上に現れることになる。

【 0 1 1 0 】

図 2 8 (b) は、『Unused』のTKIが存在しており、ここに新たなTKI、AOBファイルを書き込む場合、その書き込みがどのように行われるかを示す図である。図 2 8 (b) では、図 2 8 (a) 同様、TKI#2、TKI#4、TKI#5、TKI#7、TKI#8が『Unused』であることがわかる。

ここで4つのAOBからなる曲を書き込もうとする場合を想定する。ここでAOBの記録にどの空きTKIを割り当てるかは、後述するDPL_TK_SRPにより決定されるか、又は、任意のTKIが割り当てられる。その4つのAOBには、TrackManagerにおいて、『Unused』に設定されているTKI#2、TKI#4、TKI#7、TKI#8が割り当てられる。

【 0 1 1 1 】

これら4つのAOBは1つの曲を構成するものなので、TKI#2についてのDPL_TK_ATRを『Head_of_Song』と、TKI#4、TKI#7についてのDPL_TK_ATRを『Midpoint_of_Song』と、TKI#8についてのDPL_TK_ATRは、『End_of_Song』と設定される。曲SongDを構成する4つのTKI#2、TKI#4、TKI#7、TKI#8は、各TKI_LNK_PTRが、曲SongDを構成する次のTKI_LNK_PTRを指示するよう設定されている。即ち、矢印TL2,TL4,TL7に示すように、TKI#2のTKI_LNK_PTRはTKI#4を指示しており、TKI#4のTKI_LNK_PTRはTKI#7を、TKI#7のTKI_LNK_PTRはTKI#8を指示している。

【 0 1 1 2 】

その後、TKI#2、TKI#4、TKI#7、TKI#8のそれぞれと同じ番号を有する4つのファイルAOB002.SA1、AOB004.SA1、AOB007.SA1、AOB008.SA1が作成されて、これら4つのファイルにSongDを構成する4つのAOBが収録される。

かかるDPL_TKN、DPL_TK_ATRの設定により、4つ目の曲SongDは、TKI#2、TKI#4、TKI#7、TKI#8を用いて管理されることとなる。

【0 1 1 3】

以上のように、フラッシュメモリカード3 1に新たに曲を書き込む場合、それまでTrackManagerに『Unused』に設定されているTKIを、その新規に記録すべき曲についてのTKIに割り当てていることがわかる。

{17-5_22-18_29(a)(b)} 2つの曲を統合する場合のTKI設定
続いて曲の統合(case3)を行う際の、TKIの更新について説明する。

【0 1 1 4】

図29(a)(b)は、2つの曲を1つに統合する場合にTKIがどのように設定されるかを示す図である。図29(a)は、図19に示したTrackManagerと同一であり、図29(a)において、SongCとSongEとを1つの曲に統合するという編集操作を操作者が希望しているものとする。これらSongC、SongEに対応するAOBがAOB003.SA1、AOB008.SA1に収録されており、それらがTKI#3、TKI#8に対応づけられているので、これらTKI#3及びTKI#8のTKI_BLK_ATRの書き換えが行われる。図29(b)は、TKIのTKI_BLK_ATRの書き換え後を示す図である。本図においてTKI#3、TKI#8のTKI_BLK_ATRはSongと記載されているが、図29(b)では、TKI#3のTKI_BLK_ATRは『Head_of_Song』に書き換えられ、TKI#8のTKI_BLK_ATRは『End_of_Song』に書き換えられている。このように、TKI_BLK_ATRが書き換えられることにより、TKI#3、TKI#8、これらに対応するAOB003.SA1、AOB008.SA1は、SongCという1つの曲として扱われる。

【0 1 1 5】

ここで留意すべきは、TKIのTKI_BLK_ATRは書き換えられたが、AOB003.SA1とAOB008.SA1とを統合するという処理は行われなかった点である。何故なら、これらのAOBファイルは、互いに異なる暗号鍵にて暗号化されているので、これらを1つに統合するとなると、暗号化されたAOBファイルを復号して再度暗号化するという復号化－暗号化という2つの処理が各AOBファイルについて行う必要があり、多大な処理負荷が要求されるからである。また、統合後のAOBファイルは、1つの暗号鍵にて暗号化されるので、統合前と比較して、著作権保護の弱体化を招くから

である。

【0116】

加えてTKIは、元々TKTMSRTのサイズが大きくなるように定められているのに、編集操作においてこれを1つに統合するとすると、統合後のTKIのサイズが、大きくなり過ぎる恐れがあるからである。

以上のように、本実施形態における曲の統合化編集は、AOBファイルの暗号化を維持したまま、TKI_BLK_ATRの属性変更のみで実現されることがわかる。

【0117】

続いて曲の分割(case4)を行う際の、TKIの更新について説明する。

{17-5_22-19_30(a)(b)} 曲を分割する場合のTKI設定

図30(a)(b)は、1つの曲を2つの曲に分割する場合を想定した図である。本図におけるTrackManagerは、図27に示すTrackManagerと同一であり、本図において、SongCをSongC-SongFという2つの曲に分割するという編集を操作者が希望しているものとする。SongCをSongC-SongFに分割しようとする、SongFに対応するAOB002.SA1が生成される。図30(a)では、TKI#2が『Unused』に設定されており、『Unused』に設定されているTKI#2は、新たに生成されたAOB002.SA1に割り当てられる。

【0118】

{17-5_22-19_30(a)(b)-1_31(a)(b)}

ディレクトリエントリー及びFAT値の更新

ここでAOB003.SA1を分割して、AOB002.SA1を生成する際、ディレクトリエントリー及びFAT値を更新せねばならない。これらディレクトリエントリー及びFAT値をどのように更新するかを以下に説明する。図31(a)は、分割前において、AOB003.SA1が属するSD-AudioディレクトリについてのSD-Audioディレクトリエントリーがどのように記述されているかを示す図である。AOB003.SA1は、複数に分割されて、クラスタ007,008,009,00A……00D,00Eに格納されているものとする。この場合、ディレクトリエントリーにおけるAOB003.SA1について『ファイル最初のクラスタ番号』は、『007』と記述され、クラスタ007,008,009,00A……00Dに対応するFAT値007,008,009,00A……00Dは、それぞれ(008),(009),(00A)……(00D

), (00E)と記述されている。

【 0 1 1 9 】

この状態でAOB003.SA1の後半部を分割してAOB002.SA1を得る場合、SD-Audioディレクトリエントリーには、AOB002.SA1についての『ファイル名』、『ファイル拡張子』、『ファイル最初のクラスタ番号』が追加される。図 3 1 (b) は、分割後において、AOB003.SA1が属するSD-AudioディレクトリについてのSD-Audioディレクトリエントリーがどのように記述されているかを示す図である。

【 0 1 2 0 】

本図におけるクラスタ00Fは、操作者により指定された分割境界を含むクラスタ00Bの内容のコピーを格納したものである。クラスタ00Bに格納されているAOB002.SA1の分割部分に後続する分割部分は、クラスタ00C,00D,00E以降に格納されている。AOB002.SA1の先頭部分はクラスタ00Fに格納され、残りの部分は、クラスタ00C,00D,00E以降に格納されているので、AOB002.SA1についての『ファイル最初のクラスタ番号』には、クラスタ00Fを示すクラスタ番号00Fが記述され、クラスタ00F,00C,00D,00Eに対応づけられているFAT値00F,00C,00D,00Eには、(00C), (00D), (00E)が記述される。

【 0 1 2 1 】

{17-5_22-19_30(a)(b)-2_32(a)(b)} TKI内の情報要素の設定

以上のディレクトリエントリー及びFAT値の更新によりAOB002.SA1を得た後、AOB002.SA1についてのTKI内の情報要素をどのように設定するかについて説明する。分割された曲についてのTKIを生成する場合、TKIの情報要素には、元のTKIに記載されているものをコピーして継承すればよいもの(1)、元のTKIに基づいて更新せねばならないもの(2)の二種類が存在する。前者に該当するのは、TKTXTI_DA, ISRCであり、後者に該当するのは、BIT、TKTMSRTを初めとする残りの構成要素である。これら両者が存在するので、本実施形態では、分割された曲についてのTKIを生成する際、分割元のTKIをコピーして新たなTKIの雛型を作成すると共に、それに含まれるTKTMSRT、BITを分割・更新を行い、残りの情報要素を更新するという手順がなされる。

【 0 1 2 2 】

図 3 2 (a) は、AOB を任意の AOB_FRAME で分割する場合を想定した図である。本図において第 1 段目は、4 つの AOB_ELEMENT である AOB_ELEMENT#1、AOB_ELEMENT#2、AOB_ELEMENT#3、AOB_ELEMENT#4 を示す。これら 4 つの AOB_ELEMENT のそれぞれのデータ長は、4 つの TMSRT_entry#1,#2,#3,#4 として TKTMSRT に設定されている。本図において、AOB_ELEMENT#2 において分割境界 bd1 が設定されたとすると、AOB_ELEMENT#2 は、分割境界 bd1 より前方のフレームからなる領域 (1) と、分割境界 bd1 より後方のフレームからなる領域 (2) とに分割される。図 3 2 (b) は、AOB_ELEMENT#2 の途中部分で AOB が分割されて、AOB#1、AOB#2 という 2 つの AOB が得られた状態を示す図である。

【 0 1 2 3 】

{17-5_22-19_30(a)(b)-3_33} BIT の設定

図 3 3 は、図 3 2 に示したように AOB が分割された場合に、BIT がどのように設定されるかを示す図である。図 3 2 に示した AOB は、分割境界 bd1 にて分割されており、その分割により得られた AOB#1 は、AOB_ELEMENT#1 と、AOB_ELEMENT#2 という 2 つの AOB_ELEMENT を含み、AOB#2 は、AOB_ELEMENT#1、AOB_ELEMENT#2、AOB_ELEMENT#3 という 3 つの AOB_ELEMENT を含んでいることがわかる。

【 0 1 2 4 】

これらの AOB_ELEMENT のそれぞれにも、三角旗状の記号が付与されているが、これらは、それぞれ AOB に対応する TKI に含まれる TMSRT_entry が設定されていることを示す。先ず最初に分割により得られた AOB#1 について説明する。AOB#1 に含まれる AOB_ELEMENT#1、AOB_ELEMENT#2 は、クラスタ 007 ~ クラスタ 00A を占有しているので、AOB#1 は、クラスタ 007 ~ クラスタ 00A を一単位として扱われる。ここで AOB#1 における AOB_ELEMENT#2 は、クラスタ 00A の終端迄を占有しているのではなく、クラスタ 00A の存在する分割境界 bd1 迄を占有しているので AOB#1 についての SZ_DATA は、領域 md0 から、クラスタ 00A における分割境界 bd1 までのデータ長を指示することになる。AOB#1 の『FNS_1st_TMSRTE』は分割前と変わらないが、AOB#1 の『FNS_Last_TMSRTE』は、AOB_ELEMENT#2 の分割前の先頭から、分割境界 bd1 までのフレーム数を指示している点が分割前と異なる。

【 0 1 2 5 】

続いて分割により得られたAOB#2について説明する。AOB#2に含まれるAOB_ELEMENT#1、AOB_ELEMENT#2、AOB_ELEMENT#3は、クラスタ00B～クラスタ00Fを占有している。クラスタ00Fとは、クラスタ00Aの内容のコピーを格納しているクラスタである（クラスタ00Fにクラスタ00Aのコピーを格納している理由は、クラスタ00Aは、AOB#1のAOB_ELEMENT#2により占有されているので、このクラスタと異なるクラスタをAOB#2に含まれるAOB_ELEMENT#1に割り当てる必要があるからである。）。

【0 1 2 6】

AOB#2におけるAOB_ELEMENT#1は、クラスタ00Fの先端から占有しているのではなく、クラスタ00Fの存在する分割境界bd1以降を占有しているのでAOB#2についてのSZ_DATAは、クラスタ00Bの先頭から、クラスタ00Eにおける途中部分までのデータ長と、クラスタ00FにおいてAOB_ELEMENT#1が占有しているデータ長との和を指示することになる。

【0 1 2 7】

クラスタ00Fに格納されているクラスタ00Aのコピーには、AOB#1のAOB_ELEMENT#2が記録されており、AOB#1のAOB_ELEMENT#2により占有されている部分を、AOB#2から除外されねばならないので、AOB#2のBITについてのDATA_Offsetは、クラスタ00FにおいてAOB#1のAOB_ELEMENT#2により占有されているサイズが設定されている。

【0 1 2 8】

この図からもわかるように、AOBの分割においては、分割境界を含むAOB_ELEMENTのみが2つに分割され、その分割境界の前後のAOB_ELEMENTは、分割前のものから変化していないことがわかる。そのため、AOB#2の『FNs_Last_TMSRTE』は、分割前のAOB_ELEMENT#4の『FNs_Last_TMSRTE』と同じ値に設定され、AOB#2の『FNs_1st_TMSRTE』は、AOB#2のAOB_ELEMENT#1、即ち、分割前のAOB_ELEMENT#2における分割境界以降の終端部分に含まれるフレーム数が設定される。

【0 1 2 9】

{17-5_22-19_30(a)(b)-4_34} BITの設定

図3 4は、分割の前後でBITがどのように変化するかを更に具体的に示す図で

ある。図34の左側のBITは、分割前のBITの設定例を示す。曲を分割する前のBITは、Data_OffsetがXに設定され、SZ_DATAが『52428』、TMSRTE_Nsが『n』個と設定される。FNs_1st_TMSRTEは『80フレーム』、FNs_Middle_TMSRTEについては『94フレーム』に設定され、FNs_Last_TMSRTEは『50フレーム』に設定されることがわかる。

【0130】

図34の右側に、分割後の2つの曲についてのBITの設定を示す。本BITに対応するAOBが図32(a)に示すように分割された場合、1曲目のBITにおいて、Data_Offsetは分割前と同一値『x』に設定されるが、SZ_DATAに分割点Qまでのデータ長『Q』に更新され、TMSRTE_Nsには、1番目のTMSRT_entryからk番目のTMSRT_entryまでのTMSRT_entryの個数である『k個』に更新される。FNs_1st_TMSRTE及びFNs_Middle_TMSRTEについては分割前同様、80,94フレームに設定されるが、分割後の1曲目のAOBの最後のAOB_ELEMENTには、図32(a)においてp個のAOB_FRAMEが含まれているので、FNs_Last_TMSRTEは『pフレーム』に設定される。

【0131】

2曲目のBITは、Data_OffsetがRに設定され、SZ_DATAがオリジナルのSZ#DATA52428-分割点Qまでのデータ長、TMSRTE_Nsがn-k+1個と設定される(k番目のTMSRT_entryからn番目のTMSRT_entryまでのTMSRT_entry個数であるn-k個と、分割のために新たに追加されたk番目のTMSRT_entryの個数である1個とを加算した数である。)。FNs_Middle_TMSRTE及びFNs_Last_TMSRTEについては分割前同様、94,50フレームに設定されるが、分割後の2曲目のAOBの最初のAOB_ELEMENTには、94-p個のAOB_FRAMEが含まれているので、FNs_1st_TMSRTEは『94-pフレーム』に設定される。

【0132】

{17-5_22-19_30(a)(b)-5_35} BITの設定

図35は、分割後のTKTMSRTを示す図である。まずTMSRTについては以下のようになる。1曲目のTMSRTは分割前のAOBのTMSRTのはじめからk番目のエントリまで(TMSRT_entry#1~TMSRT_entry#k)を含む。ここで注意すべきは、分割境界を含むAOB_ELEMENT#kは、領域(1)を含むのみなので、このk番目のエントリは、この

領域(1)に相当する部分のデータサイズのみが含まれている。2曲目のTMSRTは、分割前のk番目のエントリからn番目のエントリまで(TMSRT_entry#k~TMSRT_entry#n)を含む。ここで注意すべきは、分割境界を含むAOB_ELEMENT#kは、2曲目において領域(2)を含むのみなので、分割前のk番目のエントリは、この領域(2)に相当する部分のデータサイズのみが含まれている。

【0 1 3 3】

TKIをコピーすると共に、TKTMSRT、BITを分割・更新を行い、残りの情報要素を更新すれば、分割により得られた新たな曲についてのTKIが得られることになる。ここで留意すべきは、曲を分割する場合においても、AOBファイルの暗号化を復号することなく、暗号化された状態のままAOBファイルに対応する曲を2つに分割することができる。AOBファイル分割の際に復号・再暗号化が伴わないので、曲を分割する際の処理負荷が軽減されていることがわかる。これにより、再生装置の処理性能が低い場合でも、曲の編集を行うことができる。

【0 1 3 4】

以上長文となったが、TKIについての説明を終了する。続いてプレイリストについて説明する。

{17-6} Playlistmanager

図17に示すPlaylistmanagerは、破線の引き出し線h5に示すように、フラッシュメモリカード31内に格納されているプレイリストを管理するPlaylistManager_Information(PLMGI)と、フラッシュメモリカード31に格納される全曲を管理するDefault_Playlist_Information(DPLI)と、ユーザにより自由に定義されるPlaylistInformation(PLI)#1,#2,#3,#4,#5……#mとからなり、Default_Playlist情報は、破線の引き出し線h6に示すように、Default_Playlist_General_Information(DPLGI),Default_Playlist_Track_Serch_Pointer(DPL_TK_SRP)#1,#2,#3,#4……#mからなることがわかる。また各PLIは、破線の引き出し線h7に示すように、Playlist_General_Information(PLGI),Playlist_Track_Serch_Pointer(PL_TK_SRP)#1,#2,#3,#4……#mからなることがわかる。

【0 1 3 5】

{17-7_18} プレイリストの個数、データサイズ

図 18 を参照すると、プレイリストの最大数は 99 個である。また、Playlist Manager Information (PLMGI) と Default Playlist Information (DPLI) は、合計で 560 バイトの固定長である。Playlist Information (PLI) もまた、512 バイトの固定長である。Default Playlist 情報に含まれる DPL_TK_SRP は、DPL_TK_ATR, DPL_TKN を含んでいる。一方、PlayList 情報に含まれる PL_TK_SRP は、PL_TKN のみを含んでいる。これらの DPL_TK_ATR, DPL_TKN, PL_TKN は、図 36 に示すフォーマットを有する。

【0136】

{17-8_36(a)-1} DPL_TK_SRP のフォーマット

図 36 (a) は、DPL_TK_SRP のフォーマットを示す図である。図 36 (a) において DPL_TK_SRP は、0 ビット目から 9 ビット目までに、DPL_TKN が記述され、13 ビット目から 15 ビット目までには、DPL_TK_ATR が記述され、10 ビット目から 12 ビット目までは予約用に確保 (reserved) されている。

【0137】

次に、0 ビット目から 9 ビット目までのフィールドを占める DPL_TKN には、TKI 番号が記述される。ここに TKI 番号を記述することにより、TKI を特定することが可能となる。

{17-9_36(b)} PL_TK_SRP のフォーマット

図 36 (b) は、PL_TK_SRP のフォーマットを示す図である。PL_TK_SRP は、0 ビット目から 9 ビット目までのフィールドを有しており、ここに PL_TKN、即ち、TKI 番号が記述される。

【0138】

{17-8_36(a)-2} DPL_TK_ATR の構成

図 36 (a) の『DPL_TK_ATR』から破線の矢印 h51, h52 にて引き出された枠内に、DPL_TK_ATR の設定例を示す。この枠内の記載からも理解できるように、DPL_TK_SRP についての DPL_TK_ATR の設定は、TKI についての TKI_BLK_ATR の設定と同一であり、『Song』、『Head_of_Song』、『Midpoint_of_Song』、『End_of_Song』の何れかが設定される。

【0139】

具体的には、TKINにて指定されたTKIが使用中であり、当該TKIに対応するAOBファイルに1個の曲が収録されている場合（TKIのTKI_BLK_ATRにおける『Song』）、DPL_TK_ATRは"000b"の値が設定される。

TKINにて指定されたTKIが使用中であり、当該TKIに対応するAOBファイルに曲の先頭部のみが収録されている場合（TKIのTKI_BLK_ATRにおける『Head_of_Song』）、DPL_TK_ATRは"001b"の値が設定される。

【0 1 4 0】

TKINにて指定されたTKIが使用中であり、当該TKIに対応するAOBファイルに曲の中間部のみが収録されている場合（TKIのTKI_BLK_ATRにおける『Midpoint_of_Song』）、DPL_TK_ATRには"010b"の値が設定される。

TKINにて指定されたTKIが使用中であり、当該TKIに対応するAOBファイルに曲の終端部のみが収録されている場合（TKIのTKI_BLK_ATRにおける『End_of_Song』）、DPL_TK_ATRには、"011b"の値が設定される。

【0 1 4 1】

TKINにて指定されたTKIが未使用であり、TKIの領域のみが確保されている場合、すなわち削除されたTKIである場合（TKIのTKI_BLK_ATRにおける『Unused』）、"100b"の値が設定される。

TKINにて指定されたTKIが未使用であり、TKIの領域が確保されていない場合、すなわち初期状態のTKIである場合は、"101b"の値が設定される。

【0 1 4 2】

『DPL_TK_SRP』は、DPL_TKNにTKIの番号を記述することにより、複数のTKIのうち、何れかのものとの対応関係を有する。また、Default_Playlist情報におけるDPL_TK_SRPの順位は、DPL_TK_SRPと対応関係を有するTKIに対応するAOB（AOBファイル）が何番目に再生されるかを示す。これらのことにより、Default_Playlist情報におけるDPL_TK_SRPの順序は、各TKIに対応するAOBファイルに収録されているAOBをどのような順序で再生させるか、即ち、AOBの再生順序を定義することとなる。

【0 1 4 3】

{17-9_37-1} Default_Playlist情報、TKI、AOBファイルの相互関係

図 3 7 は、Default_Playlist 情報、TKI、AOB ファイルの相互関係を示す図である。本図における第 2、第 3、第 4 段目は、図 1 9 の第 1 段目、第 2 段目、第 3 段目と同一であり、8 つの TKI を含む TrackManager、8 つの AOB ファイルを示す。図 1 9 と異なるのは、第 1 段目に Default_Playlist 情報を示す四角枠が記述されている点である。第 1 段目の枠に含まれる 8 つの小枠は、Default_Playlist 情報に含まれる 8 つの DPL_TK_SRP を示す。これらの小枠の上段は DPL_TK_ATR を示し、下段は DPL_TKN を示す。

【 0 1 4 4 】

本図における矢印 DT1, DT2, DT3, DT4……を参照すれば、DPL_TK_SRP#1 と、TKI#1 との間に対応関係が成立しており、DPL_TK_SRP#2 と、TKI#2 との間、DPL_TK_SRP#3 と、TKI#3 との間、DPL_TK_SRP#4 と、TKI#4 との間にも対応関係が成立していることがわかる。

更に、各 DPL_TK_SRP における DPL_TK_ATR を参照すれば、DPL_TK_SRP#1、DPL_TK_SRP#2、DPL_TK_SRP#3、DPL_TK_SRP#8 は何れも、Song と設定されている。即ち、DPL_TK_SRP#1 → TKI#1 (AOB001.SA1)、DPL_TK_SRP#2 → TKI#2 (AOB002.SA1)、DPL_TK_SRP#3 → TKI#3 (AOB003.SA1)、DPL_TK_SRP#8 → TKI#8 (AOB008.SA1) という 4 つの組みは、それぞれが独立した曲に対応しているのである。

【 0 1 4 5 】

DPL_TK_SRP#4、DPL_TK_SRP#5、DPL_TK_SRP#6、DPL_TK_SRP#7 の DPL_TK_ATR は何れも Song と設定されず、DPL_TK_SRP#4 における DPL_TK_ATR は『Head_of_Song』と設定され、DPL_TK_SRP#7 における DPL_TK_ATR は『End_of_Song』と、DPL_TK_SRP#5、DPL_TK_SRP#6 は『Midpoint_of_Song』と設定されていることがわかる。このことは、DPL_TK_SRP#4 と対応関係を有する TKI#4 (AOB004.SA1) が、曲の先頭部であり、DPL_TK_SRP#5、#6 と対応関係を有する TKI#5 (AOB005.SA1) 及び TKI#6 (AOB006.SA1) が、曲の中間部、DPL_TK_SRP#7 と対応関係を有する TKI#7 (AOB007.SA1) が、曲の終端部であることを意味する。

【 0 1 4 6 】

DefaultPlaylist における DPL_TK_SRP の順序は、各 TKI に対応づけられている AOB をどのような順序で再生させるかを示す。本図の DefaultPlaylist 内の DPL_TK_S

RP#1,#2,#3,#4……#8のDPL_TKNは、TKI#1,#2,#3,#4……#8を示しているので、矢印(1)(2)(3)(4)……(8)に示すようにTKI#1に対応するAOB001.SA1が1番目に再生され、TKI#2に対応するAOB002.SA1が2番目、TKI#3に対応するAOB003.SA1が3番目、TKI#4に対応するAOB004.SA1が4番目に再生されることになる。

【0 1 4 7】

{17-10_38} DefaultPlaylist、Playlist情報の設定例

図3 8は、DefaultPlaylist、Playlist情報の設定例を、図3 7と同様の表記で示した図である。本図における第1段目における四角枠はDefault_Playlist情報を示し、第2段目における3つの四角枠はPlaylist情報を示す。DefaultPlaylistに含まれる小枠は、DefaultPlaylistに含まれる8つのDPL_TK_SRPを示し、Playlist情報に含まれる小枠は、3つ又は4つのPL_TK_SRPを示す。本図のDefault_Playlist情報に含まれる各DPL_TK_SRPのTKINの設定は、図3 7と同一である。しかし、Playlist情報に含まれるPL_TK_SRPのTKINの設定は、DPL_TK_SRPのそれと全く異なることがわかる。

【0 1 4 8】

{17-10_39} DPL_TK_SRPとTKIとの対応

図3 9は、図3 7と同じ表記法を用いてDPL_TK_SRPとTKIとの対応を示す図である。図3 9においてPlaylist#1は、PL_TK_SRP#1,#2,#3からなる。このうちPL_TK_SRP#1のPL_TKNは#3と記載されており、PL_TK_SRP#2のPL_TKNは#1と、PL_TK_SRP#3のPL_TKNは#2と記載されているので、Playlist情報#1を用いて曲を再生する場合、矢印(11)(12)(13)に示すように複数のAOBはAOB#3,#1,#2の順序で再生される。

【0 1 4 9】

Playlist#2は、PL_TK_SRP#1,#2,#3からなる。このうちPL_TK_SRP#1のPL_TKNは#8と記載されており、PL_TK_SRP#2,#3のPL_TKNは#3、#1と記載されているので、Playlist情報#2を用いて曲を再生する場合、矢印(21)(22)(23)に示すように複数のAOBはAOB#8,#3,#1という順序、即ちPlaylist#1と全く異なる順序で再生される。

【0 1 5 0】

Playlist#3は、PL_TK_SRP#1,#2,#3,#4からなる。このうちPL_TK_SRP#1,#2,#3,#4のPL_TKNは#8,#4,#3,#1と記載されているので、PlayList情報#3を用いて曲を再生する場合以下に示す再生順序でAOBが再生される。先ず矢印(31)に示すようにSongEを構成するAOB#8が再生され、矢印(32)に示すようにSongDを構成するAOB#4,AOB#5,AOB#6,AOB#7がこれに続いて再生される。続いて、矢印(33)(34)に示すようにSongC、SongAを構成するAOB#3,AOB#1という順序で再生される。ここで注意すべきは、トラックが複数のTKIから構成される場合、PL_TK_SRPのエントリーには、複数TKIのうち、先頭のTKI番号のみが記述されている点である。具体的にいうと、Default_Playlist情報におけるDPL_TK_SRPは、SongDについての4つのTKIであるTKI#4、TKI#5、TKI#6、TKI#7を指定していたが、PlayList情報におけるPL_TK_SRPは、それら4つのTKIを指定する必要はない。Playlist#3のPL_TK_SRP#2がTKI#4~TKI#7のうち、TKI#4のみを指定していることは、このことを意味している。この場合再生時には、PL_TK_SRPに記述されているTKIに従って、AOBファイルを再生させてゆくだけでは足りず、DPL_TK_SRPの検索を行わねばならない。即ち、複数TKIのうち、先頭のTKI番号のみが記述されているPL_TK_SRPを用いてTKI(AOB)を再生するには、PL_TK_SRPに記述されているTKIを元にDPL_TK_SRPを検索し、トラックが複数のTKIから構成されているか否かを判定する。複数のTKIから構成されている場合には、対応するTKI(AOB)を全て再生するという手順を経るのである。

【0 1 5 1】

以上のように、PlayListManagerにはDefaultPlaylist、複数のPlayList情報が記述され、これらを構成するDPL_TK_SRP、PL_TK_SRPのDPL_TKN、PL_TKNにそれぞれ相異なる再生順序が記載されていれば、複数AOBは、それぞれ相異なる再生順序で再生されることになる。全く異なる再生順序で再生されれば、操作者は、複数の音楽アルバムが格納されているような感覚でフラッシュメモリカード31を利用することができる。

【0 1 5 2】

また、注意すべきは、何れもAOBファイルに対応づけられているDPL_TK_SRP、TKIのうち、DPL_TK_SRPのデータサイズは小さく(2バイトに過ぎない)、TKIのデー

タサイズは大きい(1024バイトもある。)点である。TrackManagerにおけるTKIの順序を入れ替えることは、フラッシュメモリカード31に対するアクセスが多く発生するが、Default_Playlist情報、PlayList情報におけるDPL_TK_SRPの順序を入れ替えても、フラッシュメモリカード31に対するアクセスはそれほど多くなならない。この点に鑑み、ナビゲーションデータは、その編集時において、編集操作に応じてDefaultPlaylistにおけるDPL_TK_SRPの順序を積極的に変化させる一方、TrackManagerにおけるTKIの順序は、編集操作にかかわらず、一定に維持するようにしている。

【0153】

{17-9_37(a)(b)-2_40(a)(b)} DPL_TK_SRPの順序を入れ替え

次に、Default_Playlist情報におけるDPL_TK_SRPの順序を入れ替えることにより、曲の再生順序を変更するという編集操作がどう行われるかについて説明する。図40(a)(b)は、曲の順序を入れ替える場合を想定した図である。図40(a)におけるDPL_TK_SRP、TKIの設定は、図37と同じである。図37(a)においてDPL_TK_SRP#3におけるDPL_TKNはTKI#3、DPL_TK_SRP#8におけるDPL_TKNはTKI#8と設定されていたが、この状態において、太枠で囲ったDPL_TK_SRP#3と、DPL_TK_SRP#8との順番を入れ替える。図40(b)における(1)(2)(3)(4)(5)(6)(7)(8)は、順番入れ替え後の曲の再生順序を示す。このことに留意すると、図40(a)における再生順序は、SongA、SongB、SongC、SongD、SongEであるが、図40(b)におけるDefault_Playlist情報では、DPL_TK_SRP#3、DPL_TK_SRP#8についてのDPL_TKNの順序が入れ替えられたので、SongA、SongB、SongE、SongD、SongCの順序で再生されることになる。このように、Default_Playlist情報における、DPL_TK_SRPの順序を入れ替えることにより、簡易に曲の再生順序を変更することができる。

【0154】

曲の変更操作という編集操作について説明したところで、TKIの場合と同様、一部の曲が削除された場合(case1)、一部の曲が削除された後、新たな曲を記録する場合(case2)、複数の曲のうち、任意の2つを1つの曲に統合する場合(case3)、1つの曲を分割して、2つの曲を得る場合(case4)において、DPL_TK_SRP及びTKI

がどのように更新されるかについて説明する。

【0 1 5 5】

{17-9_37(a)(b)-3_41(a)(b)} 曲を削除する場合

先ず初めに、一部の曲が削除された場合(case1)について説明する。

図4 1 (a) (b) は、図3 7に示したDefaultPlaylistのうち、DPL_TK_SRP#2及びTKI#2を削除する場合にDefaultPlaylist、TrackManager、AOBファイルがどのように更新されるかを示す図である。図4 1は、TKIの削除の説明で引用した図2 7と同一部分を有する。即ち、図4 1における第2、第3、第4 段目は図2 7と同一である。異なるのは図3 7同様、第1 段目に複数のDPL_TK_SRPを含むDefault_Playlist情報が記載されている点である。図4 1 (a)において太枠で囲ったDPL_TK_SRP#2→TKI#2 (AOB002.SA1) からなるSongBをユーザが削除したものとする。この場合、Default_Playlist情報においてはDPL_TK_SRP#2が削除されて、DPL_TK_SRP#3～DPL_TK_SRP#8は、DPL_TK_SRP#2が占有していたフィールドを詰めるように、順番が1つずつ繰り上がる。このように各DPL_TK_SRPの順番を繰り上がり、一番最後のDPL_TK_SRP#8が『Unused』に設定される。これに対してTKIは、図2 7 (a) (b) を用いて説明したように『Unused』に設定されているのみで、TKI#2を詰めるような移動は行われていない。またAOB002.SA1は、削除されていることがわかる。DPL_TK_SRPについては順番の繰り上げが行われたが、TKIについては順番の繰り上げが行われていないので、図4 1 (b) では、DPL_TK_SRPにおけるDPL_TKNが更新されている。即ち、新たなDPL_TK_SRP#2のDPL_TKNは、矢印DT11に示すようにTKI#3を指示しており、DPL_TK_SRP#3のDPL_TKNは矢印DT12に示すようにTKI#4を、DPL_TK_SRP#4のDPL_TKNはTKI#5、DPL_TK_SRP#5のDPL_TKNはTKI#6をそれぞれ指示している。更に、『Unused』に設定されたDPL_TK_SRP#8のDPL_TKNは、矢印DT13に示すように、『Unused』に設定されたTKI#2を設定していることがわかる。

【0 1 5 6】

曲の削除が行われた場合、使用中であるDPL_TK_SRPが先頭に繰り上げられるが、それに対応するTKIは、もとの配置を保ったまま、未使用に設定されることがわかる。このように、TKIの配置を編集前後において、不動とするので、編集処

理に伴う処理負荷を軽減することができる。

{17-9_37(a)(b)-4_42(a)(b)} 曲を記録する場合のTKIの割り当て

続いて一部の曲が削除された後、新たな曲を記録する場合(case2)について説明する。図4 2 (a) (b)は、『Unused』のTKIと、DPL_TK_SRPとが存在しており、ここに新たなTKI、DPL_TK_SRPを書き込む場合、その書き込みがどのように行われるかを示す図である。図4 2 (a) (b)において、『Unused』のTKIに新たなTKIを割り当てるケースを説明した際、引用した図2 8 (a) ~ (b)と同一部分を有する。即ち、図4 2 (a) (b)における第2、第3、第4段目は、図2 8 (a) (b)の第1、第2、第3段目と同一である。異なるのは、図4 2の第1段目に複数のDPL_TK_SRPからなるDefault_Playlist情報が記述されている点である。図4 2 (a)において、DPL_TK_SRP#4~DPL_TK_SRP#8が『Unused』であり、一方、図2 8 (a)に示したようにTKI#2、TKI#4、TKI#5、TKI#7、TKI#8が『Unused』であることがわかる。TrackManagerにおいて『Unused』のTKIが虫食い状に存在しているのに対して、Default_Playlist情報において『Unused』のDPL_TK_SRPがまとめられているのは、上述したように、DPL_TK_SRPは、『Unused』以外のDPL_TK_SRPの繰り上げが行われるのに対して、TKIは、そのような繰り上げが行われないからである。

【0 1 5 7】

ここで4つのAOBからなるSongDを書き込もうとする場合を想定する。その4つのAOBのそれぞれについてのTKIは、TrackManagerにおいて、『Unused』に設定されているTKI#2、TKI#4、TKI#7、TKI#8のそれぞれに書き込まれる。

一方、これら4つのAOBについてのDPL_TK_SRPは、Default_Playlist情報におけるDPL_TK_SRP#4~DPL_TK_SRP#7に書き込まれる。これら4つのAOBは1つの曲を構成するものなので、DPL_TK_SRP#4についてのDPL_TK_ATRは『Head_of_Song』と、DPL_TK_SRP#5、DPL_TK_SRP#6についてのDPL_TK_ATRは『Midpoint_of_Song』と、DPL_TK_SRP#7についてのDPL_TK_ATRは『End_of_Song』と設定されている。

【0 1 5 8】

また、DPL_TK_SRP#4についてのDPL_TKNはTKI#2と設定され、DPL_TK_SRP#5についてのDPL_TKNはTKI#4、DPL_TK_SRP#6についてのDPL_TKNはTKI#7、DPL_TK_SRP#7

についてのDPL_TKNはTKI#8と設定されている。

以上のようなDPL_TKN、DPL_TK_ATRの設定により、TKI#4～TKI#7は、4つ目の曲SongDとして管理されることとなる。

【0 1 5 9】

以上の処理において、『Unused』のTKIに対する書き込みが行われたが、TKI#1、TKI#2、TKI#3、TKI#4に関しては、何の変動もなされていない点は図 2 8 の場合と同様である。

{17-9_37(a)(b)-5_43(a)(b)} 曲の統合(case3)を行う場合について

続いて曲の統合(case3)を行う際の、Default_Playlist情報の更新について説明する。図 4 3 (a) (b) は、曲の統合を行う場合を想定した図である。本図は、TKIの統合処理を説明した際に引用した図 2 9 (a) (b) と同一部分を有する。即ち、図 4 3 (a) (b) における第 2、第 3、第 4 段目は、図 2 9 (a) (b) における第 1 段目、第 2 段目と同一である。差違点は、図 4 3 (a) (b) では、Default_Playlist情報が記載されており、それに含まれるDPL_TK_SRP#8が『Unused』に設定されていて、同じく『Unused』に設定されているTKI#2と対応関係を有している点である。本図において、図 2 9 に示したような曲の統合処理が、AOBファイル及びTKIに対してなされると、DPL_TK_SRP#3～DPL_TK_SRP#6の内容を1つずつずらして、太枠で囲ったDPL_TK_SRP#7の記述内容をDPL_TK_SRP#3にコピーする。TKIについては、図 2 9 に示した場合と同様の更新処理がなされる。

【0 1 6 0】

{17-9_37(a)(b)-6_44(a)(b)} 曲の分割(case4)を行う場合について

続いて曲の分割(case4)を行う際の、Default_Playlist情報の更新について説明する。

図 4 4 (a) (b) は、曲の分割を行う場合を想定した図である。本図は、TKIについての分割処理を説明した際に引用した図 3 0 (a) (b) と同一部分を有する。即ち、本図における第 2 段目、第 3 段目は、図 3 0 (a) (b) における第 1 段目、第 2 段目と同一である。差違点は、図 4 4 (a) (b) では、Default_Playlist情報が記載されており、それに含まれるDPL_TK_SRP#8が『Unused』

に設定されていて、同じく『Unused』に設定されているTKI#2と対応関係を有している点である。この状態において、図 3 0 の場合と同様に、太枠で囲ったTKI#3、AOB003.SA1を2つに分割しようとする、DPL_TK_SRP#3～DPL_TK_SRP#7の順序を一つずつ繰り下げて、Default_Playlist情報における『Unused』のDPL_TK_SRPをDPL_TK_SRP#3まで移動する。移動後のDPL_TK_SRP#3には、分割により得られたTKI#2が対応づけられる。TKI#2に対応づけられているAOB002.SA1は、元々AOB003.SA1の後半部を格納したものであるが、TKI#2に対応づけられているDPL_TK_SRP#3の前に、DPL_TK_SRP#2が存在し、このDPL_TK_SRP#2は、TKI#2-AOB002.SA1が対応づけられている。即ち、AOB002.SA1及びAOB003.SA1は、元のAOB003.SA1の後半部分、前半部分を格納しているが、これらを指定しているDPL_TK_SRP#2、DPL_TK_SRP#3は、AOB003.SA1、AOB002.SA1の順序で、これらのAOBファイルを再生するよう再生順序を指定しているので、元のAOB003.SA1の後半部分、前半部分は、DPL_TK_SRPの再生順序指定により、前半部分、後半部分の順に、再生されることとなる。

【0 1 6 1】

{17-9_37(a)(b)-7} 編集処理の応用

以上の4つの編集操作を組み合わせることにより、操作者は、様々な編集操作を行うことができる。即ち、ある曲の先頭部分にディスクジョッキーのアナウンスが入っており、これを削除したい場合、上記の曲の分割処理にて、そのアナウンス部分を一個の曲として分割し、その後、その曲を削除すれば、ディスクジョッキーのアナウンスのみを部分的に削除することができる。

【0 1 6 2】

以上でナビゲーションデータについての説明を終え、続いて、このようなナビゲーションデータ、プレゼンテーションデータを再生するために構成された再生装置について説明する。

{45-1} 再生装置の外観

図 4 5 は、本実施形態に係るフラッシュメモリカード 3 1 についての携帯型の再生装置を示す図である。本図において再生装置は、フラッシュメモリカード 3 1 が挿入される挿入口、再生、順方向サーチ再生、逆方向サーチ再生、早送り、

巻き戻し、停止等のキー操作を操作者から受け付けるためのキーパネルと、液晶ディスプレイとを有しており、通常の携帯型音響機器同様の外観を有する。キーパネルには、プレイリスト／トラックの選択を受け付けるPlaylistキー、曲の先頭へのスキップを受け付ける『|<<キー』、次曲の先頭へのスキップを受け付ける『>>|キー』、早送り、巻き戻し、順方向サーチ再生、逆方向サーチ再生を受け付ける『>>キー』、『<<キー』、フラッシュメモリカード31に静止画が格納されている場合に、静止画を表示させる操作を受け付けるDisplayキー、録音操作を操作者から受け付けるRecキー、Stereo/Monoral選択、サンプリング周波数選択を操作者から受け付けるAudioキー、ブックマークの指定を受け付けるMarkキー、曲の編集、タイトル入力を受け付けるEditキーが備えられている。

【0163】

{45-2} フラッシュメモリカード31の携帯型再生装置における改良点

このフラッシュメモリカード31の携帯型再生装置が通常の携帯型音響機器と異なるのは、以下の改良点(1)～(4)である。即ち、操作者からDefault_Playlist情報、PlayList情報、曲の指定を受け付けるために、液晶ディスプレイには、プレイリスト、曲の一覧表示がなされること(1)、また、そのように一覧表示されたプレイリスト又は曲のうち、任意のものを再生対象又は編集対象として指定させるためのキー割り当てがなされていること(2)、曲の再生進行と共に、液晶ディスプレイには、曲の再生経過時刻が表示されること(3)、タイムサーチ機能や分割編集を行う際に、再生開始時間を設定するために用いられるジョグダイヤルが設けられていること(4)である。

【0164】

{45-2_46_47} 改良点(2)の詳細

改良点(2)の詳細は以下の通りである。図46は、プレイリストの選択が行われる際の液晶ディスプレイの表示内容の一例を示す図であり、図47は、トラック(曲)の選択が行われる際の液晶ディスプレイの表示内容の一例を示す図である。

【0165】

図46における『DEFAULTPLAYLIST』『PLAYLIST#1』『PLAYLIST#2』『PLAYLIS

T#3』『PLAYLIST#4』は、フラッシュメモリカード31に格納されているデフォルトプレイリストと、4つのプレイリストを示すASCII文字列である。また、図47(a)における『TRACK#1』『TRACK#2』『TRACK#3』『TRACK#4』『TRACK#5』は、フラッシュメモリカード31に格納されているデフォルトプレイリストにて、再生順序が指定される5つのトラックを示すASCII文字列である。図46及び図47(a)にて、ハッチングを付したこれらのプレイリスト及びトラックは、再生対象又は編集対象として指定されていることを示す。このように液晶ディスプレイにプレイリストにて再生順序が規定される曲が一覧表示され、TRACK#1が再生対象に指定された状態で>>|キーの押下が行なされると、図47(b)に示すように一覧表示された複数トラックのうち、その下のTRACK#2が再生対象に指定される。TRACK#2が、再生対象に指定された状態で>>|キーの押下が行なされると、図47(c)に示すように一覧表示された複数トラックのうち、更に下段のTRACK#3が再生対象に指定される。TRACK#3が再生対象に指定された状態で|<<キーの押下が行なされると、一覧表示された複数トラックのうち、図47(d)に示すように一段上のTRACK#2が再生対象に指定される。このように>>|キー、|<<キーの押下に応じて、何れかの曲が再生対象として選択されるので、何れかの曲が再生対象として選択された際に、図47(e)に示すように再生キーが押下されれば、そのトラックの再生が開始され、Editキーが押下されれば、そのトラックが編集対象として指定される。

【0166】

{45-3_48} 改良点(4)の詳細

続いて改良点(4)の詳細について説明する。図48は、ジョグダイアルの操作例を示す図である。ジョグダイアルにより、操作者による回転操作を受け付けて、その回転量に応じて、液晶ディスプレイに表示されている再生経過時刻を増減させる。例えば図48(a)に示すように、液晶ディスプレイに再生開始時刻が『00:00:20』と表示されているものとする。この場合、図48(b)に示すように、ジョグダイアルが反時計回りに回転されたとすると、再生開始時刻は、その回転量に応じて減少して『00:00:10』となる。また図48(c)に示すように、ジョグダイアルが時計回りに回転されたとすると、再生開始時刻は、その回転量

に応じて増加して『00:00:30』となる。

【0 1 6 7】

このように再生時間時間を増減させるのは、トラックにおける任意の再生時刻を指定するためであり、ジョグダイアルの回転により、任意の再生時刻が指定され、再生キーが押下されれば、上記〔数式2〕〔数式3〕に従って指定された位置からAOBを再生する。

また、分割編集においてジョグダイアルは、任意の再生開始時間を分割境界として特定する際、分割境界を微調整するために用いられる。

【0 1 6 8】

{49-1} 再生装置の内部構成について

続いて再生装置の内部構成について説明する。図49は、再生装置の内部構成を示す図である。本図において再生装置は、フラッシュメモリカード31を接続するためのカードコネクタ1と、キーパネル、ジョグダイアルと接続されるユーザインターフェイス部2と、RAM3と、ROM4と、プレイリスト、トラックを一覧表示する一覧表示枠、再生経過時刻が表示される再生経過時刻枠を有する液晶ディスプレイ5と、液晶ディスプレイを駆動するためのLCDドライバ6と、AOBファイル毎に異なる暗号鍵を用いて、AOB_FRAMEの暗号化を解除するデ・スクランブラ7と、デ・スクランブラ7によりAOB_FRAMEのデスクランブルが行われれば、当該AOB_FRAMEのADTSヘッダを参照して、当該AOB_FRAMEを復号することにより、PCMデータを得るAACデコーダ8と、AACデコーダ8の復号により得られたPCMデータをD/A変換して、スピーカに出力するD/Aコンバータ9と、再生装置内の統合処理を行うCPU10とを備える。このハードウェア構成からも判るように、本再生装置には、TrackManager、Default_Playlist情報を処理するための新規の構成は見られない。TrackManager、Default_Playlist情報の処理のために設けられているのは、RAM3内に確保されているDPLI常駐領域11、PLI格納領域12、TKI格納領域13、FileKey格納領域14、ダブルバッファ15と、ROM4に格納されている再生制御プログラム及び編集制御プログラムである。

【0 1 6 9】

{49-2} DPLI常駐領域11

DPLI常駐領域 1 1 は、カードコネクタ 1 に接続されフラッシュメモリカード 3 1 から読み出されたDefault_Playlist情報を常駐させるために確保されている領域である。

{49_12} PLI格納領域 1 2

PLI格納領域 1 2 は、操作者により選択され、再生対象になっているPlayList情報を格納しておくために確保されている領域である。

【 0 1 7 0 】

{49-3} TKI格納領域 1 3

TKI格納領域 1 3 は、TrackManagerに含まれる複数のTKIのうち、再生対象になっているAOBファイルに対応するTKIのみを格納しておくために確保されている領域であり、TKI1個分のデータサイズを有する。

{49-4} FileKey格納領域 1 4

FileKey格納領域 1 4 は、認証領域内のAOBSA1.KEYに含まれる複数の暗号鍵のうち、再生対象になっているAOBファイルに対応する暗号鍵のみを格納しておくために確保されている領域である。

【 0 1 7 1 】

{49-5} ダブルバッファ 1 5

ダブルバッファ 1 5 は、フラッシュメモリカード 3 1 から読み出されたクラスタデータ（クラスター個当たりに格納されるデータ）を順次入力して格納するという入力処理と、格納されたクラスタデータから暗号化AOB_FRAMEを読み出して、順次デ・スクランブラ 7 に出力するという出力処理とを並列に行う場合に用いられる入出力バッファである。ダブルバッファ 1 5 は、AOB_FRAMEとしての出力が済んだクラスタが占有していた領域を順次空き領域に解放し、この空き領域を、新たに読み出されたクラスタの格納に用いるという領域確保、即ち、リングポインタを用いた巡回式の領域確保を行う。

【 0 1 7 2 】

{49-5_50_51(a)(b)} ダブルバッファ 1 5 における入出力

図 5 0 は、ダブルバッファ 1 5 におけるデータ入出力がどのように行われるかを示す図である。図 5 1 (a) (b) は、リングポインタを用いた巡回式の領域

確保がどのように行われるかを示す図である。

これらの図において左下向きの矢印は、クラスタデータの書込先アドレスについてポインタ、即ち、書込先ポインタを示す。左上向きの矢印は、クラスタデータの読出先アドレスについてのポインタ、即ち、読出先ポインタを示す。これらのポインタは、リングポインタとして用いられる。

【 0 1 7 3 】

{49-6_50} フラッシュメモリカード 3 1 がカードコネクタ 1 に接続されると、このフラッシュメモリカード 3 1 のユーザ領域におけるクラスタデータは、矢印w1,w2に示すようにフラッシュメモリカード 3 1 から読み出される。読み出されたクラスタデータは、ダブルバッファ 1 5 において、書込先ポインタWP1,WP2に示される位置に順次格納されてゆく。

【 0 1 7 4 】

{49-7_51(a)} このように格納されたクラスタデータに含まれるAOB_FRAMEのうち、読出先ポインタ①②③④⑤⑥⑦⑧⑨に指示される位置に存在するAOB_FRAMEは、矢印r1,r2,r3,r4,r5……に示すように順次デ・スクランブラ 7 へと出力されてゆく。ここでダブルバッファ 1 5 にクラスタデータ002,003が格納されており、読出先ポインタにて示さる読出先が図 5 0 の①②③④に示すように移動して、⑤に達した場合、クラスタ002に含まれるAOB_FRAMEは全て読み出されたことになるので、新たにクラスタ004を読み出して、図 5 1 (a) の矢印w6に示すように、クラスタ002が占有していた領域に上書きする。

【 0 1 7 5 】

{49-8_51(b)} また読出先ポインタにて示さる読出先が⑥⑦に示すように移動して、⑨に達すれば、クラスタ003に含まれるAOB_FRAMEは全て読み出されたことになるので、新たにクラスタ005を読み出して、図 5 1 (b) の矢印w7に示すように、クラスタ003が占有していた領域に上書きする。以上のような、AOB_FRAMEの出力と、クラスタデータの上書きとが何度も繰り返されて、AOBファイルに含まれるAOB_FRAMEが順次デ・スクランブラ 7、AACデコーダ 8 に出力されてゆく。

【 0 1 7 6 】

{49-9_52~55} ROM 4 に格納されている再生制御プログラム

続いてROM 4 に格納されている再生制御プログラムについて説明する。

図 5 2 は、AOBファイル読み出し処理の処理手順を示すフローチャートであり、図 5 3、図 5 4、図 5 5 は、AOB_FRAME出力処理の処理手順を示すフローチャートである。

【 0 1 7 7 】

{49-9_52-1} これらのフローチャートにおいて、変数 z とは、ユーザ領域に記録されているそれぞれのAOBファイルと、それに対応するTKIと、それに含まれるAOBとを一意に指示するための変数であり、変数 y とは、変数 z にて指示されるAOB# z に含まれる個々のAOB_ELEMENTを指示するための変数である。変数 x とは、変数 y にて指示されるAOB_ELEMENT# y に含まれるそれぞれのAOB_FRAMEを指示する変数である。先ず図 5 2 を参照しながらAOBファイル読出処理の処理手順について説明する。

【 0 1 7 8 】

{49-9_52-2} ステップ S 1 においてCPU 1 0 はPlayListManagerを読み出して、Default_Playlist情報及びPlayList情報を一覧表示する。ステップ S 2 においてCPU 1 0 は、Default_Playlist情報、PlayList情報の何れに従ってAOBを再生させるかの指定を待つ。ここで、Default_Playlist情報が指定された場合、ステップ S 2 からステップ S 3 に移行して、変数 z を初期化し ($z \leftarrow 1$)、ステップ S 4 では、Default_Playlist情報におけるDPL_TK_SRP# z に対応づけられたDPL_TKN(以下これをDPL_TKN# z という)により指定されているTKI# z を特定して、そのTKI# z のみをTKI格納領域 1 3 に読み出す。そして、ステップ S 5 においてTKI# z と同じ番号を有するAOBファイル# z を特定する。ここまでの手順で、ようやく、再生すべきAOBファイルが特定されたことになる。特定されたAOBファイルは、暗号化されているので、このAOBファイルの暗号化を解除すべく、以降ステップ S 6、ステップ S 7 の処理を行う。即ち、ステップ S 6 では、認証領域をアクセスして、暗号鍵格納ファイルにおいて当該AOBファイル# z と同じ番号を有するFile Key Entry# z に格納されている暗号鍵# z を読み出す。ステップ S 7 においてCPU 1 0 は、暗号鍵# z をデ・スクランブラ 7 に設定する。かかる設定により、暗号鍵はデ・スク

ランブラ 7 に設定されたので、以降、AOB ファイルに含まれる AOB_FRAME が順次デ
・スクランブラ 7 に投入されれば、AOB_FRAME は順次再生されることになる。

【 0 1 7 9 】

{49-9_52-3} 以降、AOB ファイルを格納している各クラスタを順次読み出し
てゆく。ステップ S 8 では、ディレクトリエントリーにおけるその AOB ファイル #
z についての『ファイル最初のクラスタ番号』を特定し、ステップ S 9 において C
PU 1 0 は、そのクラスタに格納されているデータをフラッシュメモリカード 3 1
から読み出す。ステップ S 1 0 では、FAT 値にクラスタ番号が FFF と記述されてい
るか否かを判定し、もし FAT 値が FFF 以外の値なら、ステップ S 1 1 においてその
FAT 値にて指示されているクラスタに格納されているデータを読み出す。かかる
読み出し後、ステップ S 1 0 に移行する。ここで、何れかのクラスタに格納され
ているデータを読み出し、そのクラスタに対応づけられている FAT 値を参照した
際、その FAT 値に FFF 以外の何れかのクラスタ番号が記述されている限り、ステッ
プ S 1 0 - ステップ S 1 1 の処理は繰り返し行われる。これにより、その FAT 値
により指示されているクラスタが順次読み出されてゆく。その FAT 値にクラスタ
番号が FFF と記述されている場合、AOB ファイル #z を構成するクラスタは全て読み
出されたことになるので、ステップ S 1 0 からステップ S 1 2 に移行する。

【 0 1 8 0 】

{49-9_52-4} ステップ S 1 2 において CPU 1 0 は、変数 #z が DPL_TK_SRP の総
数と一致したか否かを判定する。一致しないなら、ステップ S 1 3 に移行して、
変数 #z をインクリメントした後 ($\#z \leftarrow \#z + 1$)、ステップ S 4 に移行する。ステッ
プ S 4 において Default_Playlist 情報における DPL_TK_SRP #z の DPL_TKN #z により
指定されている TKI #z を特定して、その TKI #z のみを TKI 格納領域 1 3 に読み出す
。この際、TKI 格納領域 1 3 にはそれまで使用されていた TKI が格納されているが
、CPU 1 0 は、TKI 格納領域 1 3 に既に格納されている TKI を新たに読み出した TKI
を用いて上書きする。このような上書きにより TKI 格納領域 1 3 には、最新の TKI
のみが格納されることになる。このように TKI が上書きされれば、ステップ S 5
～ステップ S 1 2 の処理を AOB ファイル #z について繰り返す。ステップ S 5 ～ス
テップ S 1 2 の処理が繰り返され、Default_Playlist 情報に含まれる全ての DPL_

TK_SRPに対応するTKI、AOBファイルが読み出されれば、変数#zとDPL_TK_SRPの総数とが一致して、ステップS 1 2がYesとなり、本フローチャートを終了する。

【0 1 8 1】

{49-9_53_54_55} AOB_FRAME出力処理

かかるAOBファイル読出処理と並行して、CPU 1 0は、図5 3、図5 4、図5 5のフローチャート従い、AOB_FRAME出力処理を行う。本フローチャートにおいてplay_timeとは、これまで再生が経過した時間、即ち、再生経過時刻を示す変数であり、液晶ディスプレイ5の時刻表示枠内の時刻は、このplay_timeの更新に応じて、表示内容が書き換えられる。また、play_dataとは、これまで再生されたデータ長である。

【0 1 8 2】

{49-9_53-1} ステップS 2 1において、CPU 1 0は、AOBファイル#zについてのクラスタデータがダブルバッファ15に蓄積されたかを監視している。クラスタデータが蓄積されない限り、このステップS 2 1を繰り返し行うが、クラスタデータが蓄積されれば、ステップS 2 2において、#x、#yの初期化を行い（#x←1, #y←1）、その後、ステップS 2 3において、AOBファイル#zについてのクラスタにおいて、TKI#zに含まれるBIT#zのData_Offset以降からAOB_ELEMENT#yにおけるAOB_FRAME#xを検出する。ここでSZ_DATAから7バイトは、ADTSヘッダが占有しているものとして、当該ADTSヘッダを参照し、ADTSヘッダに示されているデータ長が本体部のオーディオデータであると解析して、当該ADTSヘッダと、本体部のオーディオデータとを読み出し、デ・スクランブラ7に出力する。デ・スクランブラ7によりAOB_FRAMEの暗号化が解除され、AACデコーダ8により復号が行われれば、音声として再生されることになる。

【0 1 8 3】

{49-9_53-2} 検出後、ステップS 2 4において、AOB_FRAME#xをデ・スクランブラ7に出力し、ステップS 2 5において、再生経過時刻play_timeをAOB_FRAME#xに相当する再生時間だけインクリメントし、再生済みデータ数play_dataをAOB_FRAME#xに相当するデータ数だけインクリメントする。ここでAOB_FRAMEの再生時間長は、20msecであるので、再生経過時刻play_timeには、20msecが加算され

ることになる。

【0184】

1番目のAOB_FRAMEがデ・スクランブラ7に出力されれば、ステップS26においてAOB_FRAME#xのADTSヘッダを参照して、次のAOB_FRAMEが何処に存在するかを特定する。ステップS27では、変数#xのインクリメントを行い($\#x \leftarrow \#x + 1$)、次のAOB_FRAMEをAOB_FRAME#xとする。ステップS28においてAOB_FRAME#xをデ・スクランブラ7に投入する。その後、ステップS29では、play_timeを、AOB_FRAME#xに相当する再生時間だけインクリメントすると共に、play_dataをAOB_FRAME#xに相当するデータ数だけインクリメントする。AOB_FRAME#xをインクリメントした後、ステップS30においてCPU10は、#xが『FNS_1st_TMSRTE』に達したかを判定する。#xが『FNS_1st_TMSRTE』に達しないのなら、ステップS31において、Playキー以外のキーが押下されたかどうかを確認した後、ステップS26に移行する。以降、#xが『FNS_1st_TMSRTE』に達するまで、または、Playキー以外のキーが押下されるまで、ステップS26～ステップS31の処理は繰り返し行われる。ここでPlayキー以外のキーが押下された場合、本フローチャートを終了して、押下されたキーに該当する処理を行う。押下されたキーが停止キーなら再生処理を停止し、押下されたキーが一時停止キーなら一時停止を行う。

【0185】

{49-9_54-1} 一方、#xが『FNS_1st_TMSRTE』に達したなら、ステップS30がYesとなり、図54のステップS32に移行する。ステップS26からステップS30までの処理にて、AOB_ELEMENTに含まれる全てのAOB_FRAMEがデ・スクランブラ7に投入されたので、次のAOB_ELEMENTに処理対象を移行すべく、ステップS32において#yをインクリメントすると共に、#xを初期化する($\#y \leftarrow \#y + 1$, $\#x \leftarrow 1$)。

【0186】

その後、ステップS33においてTKTMSRTを参照してAOB_ELEMENT#yについての先頭アドレスを算出する。以降、ステップS34～ステップS42からなる処理を行う。ステップS34～ステップS42の処理は、AOB_ELEMENTに含まれるAOB_FRAMEを次々と読み出す処理である点で、ステップS24～ステップS31から

なる処理と同一であるといえる。ステップ S 2 4 ～ステップ S 3 1 の処理と異なるのは、後者におけるループ処理の終了条件は、#x が『FNS_1st_TMSRTE』に達することであるのに対し、前者におけるループ処理の終了条件は、#x が『FNS_Middle_TMSRTE』に達することである。#x が『FNS_Middle_TMSRTE』に達して、ステップ S 3 4 ～ステップ S 4 2 からなるループ処理が終了すると、ステップ S 4 1 が Yes となってステップ S 4 3 に移行する。ステップ S 4 3 において CPU 1 0 は #y をインクリメントすると共に、#x を初期化する ($\#y \leftarrow \#y + 1, \#x \leftarrow 1$)。その後、ステップ S 4 4 において、変数 y が、TKI #z の TMSRT_Header における (TotalTMSRT_entry_Number-1) と等しい値に達したかを判定する。#y が (TotalTMSRT_entry_Number-1) より小さい場合、AOB_ELEMENT #y は未だ、最後の AOB_ELEMENT に達してしていないので、ステップ S 4 4 からステップ S 3 2 に移行することにより、ステップ S 3 2 ～ステップ S 4 2 の処理を継続して行う。#y が (TotalTMSRT_entry_Number-1) に達した場合、最後の AOB_ELEMENT の 1 つ前の AOB_ELEMENT まで、AOB_FRAME の読み出し処理は完遂したと考えられるので、ステップ S 4 4 が Yes となり、図 5 5 のステップ S 4 5 に移行する。

【 0 1 8 7 】

{49-9_54-2} ステップ S 4 5 ～ステップ S 5 4 の処理は、最後の AOB_ELEMENT に含まれる複数の AOB_FRAME をそれぞれ読み出す処理であるという点において、上述したステップ S 3 3 ～ステップ S 4 2 の処理と同一といえる。異なるのは、ステップ S 3 3 ～ステップ S 4 2 におけるループ処理は、ステップ S 4 1 において #x が『FNS_Middle_TMSRTE』に達することがループ処理の終了条件であったのに対して、ステップ S 5 3 では、#x が『FNS_Last_TMSRTE』であり、かつ、これまで読み出されたデータサイズを示す Play_data が SZ_DATA に達することが、ループ処理の終了条件になっている点である。

【 0 1 8 8 】

このステップ S 5 3 の条件が満たされるまで、ステップ S 4 9 ～ステップ S 5 4 の処理は繰り返し行われ、この条件が満たされれば、ステップ S 5 3 が Yes となって、ステップ S 5 5 に移行する。ステップ S 5 5 において CPU 1 0 は、#z をインクリメントしてからステップ S 2 1 に移行して ($\#z \leftarrow \#z + 1$)、次の AOB フ

イルがダブルバッファ 1 5 に蓄積されるのを待つ。蓄積されれば、ステップ S 2 1 からステップ S 2 2 に移行し、次の AOB ファイルについて、ステップ S 2 2 ～ステップ S 5 4 の処理を繰り返し行う。即ち、次の DPL_TK_SRP の DPL_TKN により指定されている TKI を特定し、その TKI に対応する AOB ファイル、即ち、TKI と同じ番号を有する AOB ファイルを特定する。その後、認証領域をアクセスして、暗号鍵格納ファイルにおいて当該 TKI と同じ番号を有する暗号鍵を特定し、当該暗号鍵を読み出して、当該暗号鍵をデ・スクランブラに設定してから、その TKI と同じ番号を有する AOB ファイルに含まれる AOB_FRAME を順次読み出して再生してゆくのである。

【 0 1 8 9 】

{49-9_54-3_56} 再生経過時刻の更新

図 5 6 は、液晶ディスプレイ 5 の時刻表示枠に表示される再生経過時刻が、変数 Play_Time の更新したがい、増加してゆく様子を示す図である。本図 (a) では、再生経過時刻は 00:00:00.000 であるが、AOB_FRAME#1 の再生が終了した時点で、再生経過時刻に AOB_FRAME の時間長 20msec が加算されて 00:00:00.020 に更新されている。AOB_FRAME#2 の再生が終了した時点で、再生経過時刻に AOB_FRAME の時間長 20msec が加算されて 00:00:00.040 に、AOB_FRAME#6 の再生が終了した時点で、再生経過時刻は 00:00:00.120 となっていることがわかる。

【 0 1 9 0 】

以上が AOB_FRAME 出力処理の全貌である。本フローチャートのステップ S 3 1 において、Play キー以外のキーの押下時に、本フローチャートの処理を中断することは既に述べた通りであり、そのような Play キー以外のキーとして一時停止キーや停止キーがあることも説明済みであるが、一時停止キーや停止キー以外にも、特殊再生を再生装置に行わせるためのキーが押下された場合も、図 5 3、図 5 4、図 5 5 のフローチャートの処理は中断し、その押下されたキーに応じた処理が実行される。以降、>> キーが押下され、順方向サーチ再生を実行する場合の CPU 1 0 の処理手順と、一時停止キーや停止キーが押下された後に、ジョグダイヤルが操作されることにより、タイムサーチ機能が実行される場合の CPU 1 0 の処理手順とについて説明する。

【0191】

{49-10_57} 順方向サーチ再生

図57は、順方向サーチ再生を行う場合のCPU10の処理手順を示すフローチャートである。>>キーが操作者により押下されて図53、図54、図55のステップS31、ステップS42、ステップS54がYesになった際、CPU10により実行されるのが本フローチャートである。

【0192】

ステップS61において、CPU10はAOB_ELEMENT#yのAOB_FRAME#xから $x+f(t)-1$ までをデ・スクランブラ7に投入する。ここで、『t』とは、間欠再生時間であり、 $f(t)$ を、間欠再生時間に相当するフレーム数、 $d(t)$ を間欠再生時間に相当するデータ数とすると、ステップS62では、再生経過時刻を示すplay_time、再生済みデータ数を示すplay_dataを、 t :間欠再生時間、 $f(t)$:間欠再生時間に相当するフレーム数、 $d(t)$:間欠再生時間に相当するデータ数に基づいて更新する($x \leftarrow x+f(t)$ 、 $play_time \leftarrow play_time+t$ 、 $play_data \leftarrow play_data+d(t)$ 尚、一般に間欠再生時間は240ミリ秒(12個のAOB_FRAMEの再生時間長)に相当する。)。

【0193】

{49-10_57-1_58(a)(b)} 図58(a)(b)は、順方向サーチ再生時において、再生経過時刻がインクリメントされてゆく様子を示す図である。図58(a)は、再生経過時刻の初期状態を示し、再生時点は、AOB_ELEMENT#51のAOB_FRAME#1であることを示す。この場合の再生経過時刻は、00:00:01.000であることがわかる。ここで、間欠再生時間として、1番目から12番目までのAOB_FRAMEがデ・スクランブラ7に投入されて、再生経過時刻に1AOB_FRAMEの時間長である240ミリ秒が加算されると、図58(b)に示すように、再生経過時刻は、00:00:01.240となる。

【0194】

{49-10_57-2} これらを更新した後、ステップS63においてCPU10は、インクリメント後のAOB_FRAME#xと、AOB_ELEMENT#yの総フレーム数との大小比較して、インクリメント後のAOB_FRAME#xがAOB_ELEMENT#y内に存在するかを判定する。具体的には、AOBの先頭に位置するAOB_ELEMENTのフレーム数は『FNs_1st_TMSR

TE』であり、中間のもののフレーム数は『FNS_Middle_TMSRTE』、最後のもののフレーム数は『FNS_Last_TMSRTE』に示されるので、これらと、AOB_FRAME#xとを比較することにより、判定を行う。もし更新後のAOB_FRAME#xがAOB_ELEMENT内に存在しないのなら、ステップS 6 4においてAOB_ELEMENT#yに後続するAOB_ELEMENTが存在するかを判定する。ここでAOB_ELEMENT#yが最後のAOB_ELEMENTであり、後続するAOB_ELEMENTが存在しない場合、ステップS 6 4がNoとなり、本フローチャートの処理を終了するが、後続するAOB_ELEMENTが存在する場合、ステップS 6 5において、AOB_FRAME#xからAOB_ELEMENT#yにおけるフレーム数を減じ、ステップS 6 6において#yを更新することにより($y \leftarrow y+1$)、AOB_FRAME#xを後続するAOB_ELEMENT#yにおけるAOB_FRAMEのフレーム位置に変換する。もし更新後のAOB_FRAME#xがAOB_ELEMENT内に存在するのなら、これらステップS 6 5～ステップS 6 6をスキップして、ステップS 6 7に移行する。

【0 1 9 5】

{49-10_57-3} 続いて、間欠スキップ間隔に応じて、AOB_FRAME#x、再生経過時刻play_time、再生済みデータ数play_dataの更新を行う。ここで、間欠スキップ間隔に相当する時間をskip_time(2秒)とし、間欠スキップ間隔skip_timeに相当するフレーム数を $f(\text{skip_time})$ 、間欠スキップ間隔skip_timeに相当するデータ数 $d(\text{skip_time})$ とすると、ステップS 6 7において、これらを用いてAOB_FRAME#x、再生経過時刻play_time、再生済みデータ数play_dataを更新する($x \leftarrow x + f(\text{skip_time}), \text{play_time} \leftarrow \text{play_time} + \text{skip_time}, \text{play_data} \leftarrow \text{play_data} + d(\text{skip_time})$)。

【0 1 9 6】

{49-10_57-4_58(c)} 図5 8 (c)に示すように、AOB_ELEMENT#51内のフレーム位置を示すAOB_FRAME#xに間欠スキップ間隔を加算したものとする。この加算後の#xがAOB_ELEMENT#51のフレーム数を上回れば、AOB_ELEMENTを次のAOB_ELEMENTに更新すると共に、加算後の#xから、AOB_ELEMENT#51のフレーム数を減じることにより、AOB_FRAME#xを、AOB_ELEMENT#52におけるフレーム位置に変換する。この場合、AOB_ELEMENT#yがAOB_ELEMENT#52となり、再生経過時刻は、00:00:01.240に2.000を加算することにより、00:00:03.240となる。AOB_FRAME#xは、AOB_E

LEMENT#52におけるAOB_FRAME#62($= (3240\text{msec} - 2000\text{msec}) / 20\text{msec}$)となる。

【 0 1 9 7 】

{49-10_57-5_58(d)} その後、AOB_ELEMENT#52におけるAOB_FRAME#62がデ・スクランブラ7に投入されれば、図58(d)に示すように再生経過時刻は、00:00:03.240に0.240を加算することにより、00:00:03.480となる。

ステップS67において間欠的スキップ時間に応じた更新を行えば、ステップS68～ステップS71の処理を行う。このステップS68～ステップS71の処理は、ステップS63～ステップS66の処理と同一であり、間欠スキップ間隔skip_timeに相当するフレーム数が加算された後のAOB_FRAMEがAOB_ELEMENT#y内に存在するかどうかの判定がなされて、存在するのなら、その次のAOB_ELEMENTをAOB_ELEMENT#yとし、AOB_FRAME#xを、新たなAOB_ELEMENT#yにおけるフレーム位置に変換する。

【 0 1 9 8 】

間欠再生時間及び間欠的スキップ時間に応じて、AOB_FRAME#x、AOB_ELEMENT#yがインクリメントされれば、ステップS72において、CPU10は、TKTMSRTを参照してAOB_ELEMENT#yについての先頭アドレスを算出し、ステップS73においてAOB_ELEMENT#yにおける先頭アドレスからADTSヘッダの探索を開始することにより、AOB_FRAME#xを検出する。そして、ステップS74において、順方向スキップキー以外のキーが押下されたか否かを判定した後、ステップS61においてAOB_ELEMENT#yのAOB_FRAME#xから $x+f(t)-1$ までをデ・スクランブラ7に投入し、再度ステップS62～ステップS73の処理を繰り返す行う。

【 0 1 9 9 】

以上の処理にて、AOB_FRAME#x、AOB_ELEMENT#yがインクリメントされ、再生位置が進行してゆく。その後、操作者により再生キーが押下されれば、ステップS74がNoとなり、本フローチャートの処理を終了する。

{49-11} タイムサーチ機能の実行

タイムサーチ機能が行われた場合の処理について説明する。Default_Playlist情報における曲を一覧表示し、任意の曲の指定を受け付ける。曲が指定され、ジョグダイヤルが操作されると、再生開始時刻を更新する。再生開始時刻が増減し

た後、再生キーが押下されると、その再生が指定された時刻を Jmp_Entry (秒)と特定する。一方、指定された曲が複数の AOB からなるか、単一の AOB からなるかを判定する。単一の AOB からなる場合、{数式 2} を満たす $\text{AOB_ELEMENT}\#y$ と、 $\text{AOB_FRAME}\#x$ とを算出する。{数式 2} を満たす $\text{AOB_ELEMENT}\#y$ 、 $\text{AOB_FRAME}\#x$ が算出されれば、この AOB に対応する TKTMSRT において、 $y+2$ 番目に位置するアドレスから、 $\text{AOB_FRAME}\#x$ の探索を始め、 x 番目の AOB_FRAME が探索されれば、この x 番目の AOB_FRAME から再生を開始する。

【0 2 0 0】

{49-12}

複数の AOB からなる場合、{数式 3} を満たす $\text{AOB}\#n$ 、 $\text{AOB_ELEMENT}\#y$ と、 $\text{AOB_FRAME}\#x$ とを算出する。{数式 3} を満たす $\text{AOB}\#n$ 、 $\text{AOB_ELEMENT}\#y$ 、 $\text{AOB_FRAME}\#x$ が算出されれば、この $\text{AOB}\#n$ に対応する TKTMSRT において、 $y+2$ 番目に位置するアドレスから、 $\text{AOB_FRAME}\#x$ の探索を始め、 x 番目の AOB_FRAME が探索されれば、この x 番目の AOB_FRAME から再生を開始する。

【0 2 0 1】

続いて BIT における FNs_1st_TMSRTE は 80 フレームであり、 FNs_Last_TMSRTE は 50 フレーム、 FNs_Middle_TMSRTE が 94 フレームである AOB において、任意の再生時刻から再生を開始する場合について説明する。

{49-13_59(a)(b)}

ここで、タイムサーチ機能が行われる場合の具体例として、ジョグダイヤルにより、再生開始時刻が指定された場合に再生を開始すべき AOB_ELEMENT 、再生を開始すべきフレーム位置をどのように特定するかについて説明する。図 5 9 は、タイムサーチ機能が行われる場合の具体例を示す図である。ここで図 5 9 (a) に示すように、再生装置が把持されて、ある AOB が再生対象として指定された状態で、その右手の親指により、ジョグダイヤルの回転操作がなされ、再生開始時刻 = 00:04:40.000 (= 280.00sec) が指定されたものとする。この AOB について TKI 内の BIT が、図 5 9 (b) に示す内容である場合、再生開始時刻 = 00:04:40.000 (= 280.00sec) を {数式 2} に適用すると、

$$280\text{sec} = (\text{FNs_1st_TMSRTE} + \text{FNs_middle_TMSRTE} \cdot y + x) \times 20\text{msec}$$

$$=(80 + 94 \cdot 148 + 8) \times 20 \text{msec}$$

となるので、{数式 2} を満たす AOB_ELEMENT#y、AOB_FRAME#x として、y=148、x=8 の AOB_FRAME が得られる。

【 0 2 0 2 】

このように y=148 と特定されたので、y+2 番目の AOB_ELEMENT#150 (=148+2) のエントリーアドレスを TKTMSRT から取得して、ここから 8 番目の AOB_FRAME から、再生を開始すれば、再生経過時刻=00:04:40.000 (=280.00sec) から、再生を開始することができる。

{49-14_60_61_62}

以上で Play キーが押下された場合の、CPU 10 の処理内容の説明を終える。続いて ROM に格納されている編集制御プログラムについて説明する。本編集制御プログラムは、Edit キーが押下された場合に実行されるものであり、図 60、図 61、図 62 にその処理手順を示す。以降、これらのフローチャートを参照しながら、編集制御プログラムの処理内容について説明する。

【 0 2 0 3 】

{49-14_60-1} 編集制御プログラム

Edit キーが押下されれば、図 60 のステップ S101 において削除、分割、統合といった典型的な編集操作の何れを行うかを操作者に提示する対話画面を表示し、その後、ステップ S102 において、対話画面に対する処理が指定されたかを判定する。ここで対話画面の操作において、>>| キー、|<< キーをそれぞれ上下カーソル操作を受け付けるためのキー、即ち、上下カーソルキーとして用いるものとする。削除処理が指定されると、ステップ S103、ステップ S104 からなるループ処理に移行する。ステップ S103 では、>>| キー、|<< キーが押下されたか否かを判定し、ステップ S104 では、編集キーが押下されたか否かを判定する。>>|、|<< キーが押下されれば、ステップ S103 からステップ S105 に移行して、指示されたトラックを編集対象として指定する。一方、編集キーが押下されれば、削除すべき曲が特定されたとして、図 41 に示した処理を行い、指定されたトラックについての TKI の TKI_BLK_ATR を『Unused』に設定することにより、特定された曲を削除する。

【0204】

{49-14_60-2} 統合編集処理

統合編集が指定されると、ステップS102からステップS107～ステップS108からなるループ処理に移行する。ステップS107～ステップS108からなるループ処理では、>>|キー、|<<キーの押下と、編集キーの押下とを受け付ける。>>||、|<<キーが押下されれば、ステップS107からステップS109に移行して、指示されたトラックを編集対象として指定する。編集キーが押下されれば、ステップS108がYesとなり、ステップS110に移行する。ステップS110では、カーソルキーにて指示されたトラックを、1曲目の編集対象として指定する。その後、ステップS111～ステップS112からなるループ処理に移行する。ステップS111～ステップS112からなるループ処理では、ステップS107～ステップS108同様、>>|キー、|<<キーの押下と、編集キーの押下とを受け付ける。>>||、|<<キーが押下されれば、ステップS111からステップS113に移行して、指示されたトラックを編集対象として指定する。編集キーが押下されれば、ステップS112がYesとなり、ステップS114に移行する。ステップS114では、カーソルキーにて指示されたトラックを、2曲目の編集対象として指定する。2つの曲が特定されれば、これらに対応づけられたTKI、DPL_TK_SRPに対して、図43に示した操作を行い、TKIのTKI_BLK_ATTRを書き換えることにより、それら操作対象として選択された複数の曲を1つの曲に統合する。

【0205】

{49-14_61-1} 曲の分割処理

曲の分割が指定されると、ステップS102からステップS116～ステップS117からなるループ処理に移行する。ステップS116～ステップS117からなるループ処理では、>>|キー、|<<キーの押下と、編集キーの押下とを受け付ける。>>||、|<<キーが押下されれば、ステップS116からステップS118に移行して、指示されたトラックを編集対象として指定する。編集キーが押下されれば、ステップS117がYesとなり、ステップS119に移行する。ステップS119では、カーソルキーにて指示されたトラックを、編集対象として指

定する。その後、ステップ S 1 2 0 では、分割が指定された曲の再生を開始して、ステップ S 1 2 1 において Mark キーの押下を受け付ける。Mark キーの押下が行われると、曲の再生を一時停止し、ステップ S 1 2 2 ～ステップ S 1 2 3 からなるループ処理に移行する。ステップ S 1 2 2 では、ジョグダイヤルに対する回転操作を受け付け、ジョグダイヤルに対して回転操作がなされると、ステップ S 1 2 4 においてその回転操作に伴い、再生開始時間を増減させる。その後、ステップ S 1 2 2 ～ステップ S 1 2 3 からなるループ処理に再度移行する。回転操作により再生開始時刻が増減された状態で、編集キーが押下されれば、ステップ S 1 2 3 からステップ S 1 2 5 に移行し、ステップ S 1 2 5 において、編集キーが押下された再生時間を分割境界として指定する（尚、分割境界の指定にあたっては、アンドゥ機能（編集の取り消し）が可能である。）。その後、ステップ S 1 2 6 において図 4 4 で説した処理を行い、DPLI、TKI を更新することにより、曲の分割を行う。

【 0 2 0 6 】

{49-14_62-1} プレイリストの設定編集処理

プレイリストの設定編集が指定されると、図 6 2 のフローチャートに移行する。本フローチャートにおいて変数 k とは、これから設定されるプレイリストにより、再生順序が規定される個々のトラックを指示するための変数であり、図 6 2 のフローチャートでは、先ずステップ S 1 3 1 においてこの変数 k に 1 を設定した後、ステップ S 1 3 2 ～ステップ S 1 3 4 からなるループ処理に移行する。ステップ S 1 3 2 ～ステップ S 1 3 4 からなるループ処理では、>>| キー、|<< キーの押下と、編集キーの押下と、停止キーの押下とを受け付ける。>>| キー、|<< キーが押下されれば、ステップ S 1 3 2 からステップ S 1 3 5 に移行して、>>| キー、|<< キーにより指示されたトラックを指定する。編集キーが押下されれば、ステップ S 1 3 3 が Yes となり、ステップ S 1 3 6 に移行する。ステップ S 1 3 6 では、編集キーにて指示されたトラックを、k 番目に再生されるべきトラック（曲）として特定する。その後、ステップ S 1 3 7 において、変数 k をインクリメントして、ステップ S 1 3 2 ～ステップ S 1 3 4 のループ処理に移行する。このような処理を繰り返すことにより、2 曲目、3 曲目、4 曲目のトラック（曲）が順

次特定される。このようにして、新たに作成されたプレイリストにて再生されるべき複数の曲（トラック）が特定された状態で停止キーが押下されると、ステップ S 1 3 4 からステップ S 1 3 8 に移行して、これらに対応づけられた TKI を特定する PL_TK_SRP からなる PlayList 情報を生成する。

【 0 2 0 7 】

（記録装置）

{63-1} 記録装置

続いて、フラッシュメモリカード 3 1 の記録装置についての一例を説明する。図 6 3 は、フラッシュメモリカード 3 1 の記録装置の一例を示す図である。本図における記録装置は、インターネットを介した通信が可能であり、電子音楽配信により SD-Audio ディレクトリイが暗号化された状態で通信回線を介して伝送されてくる場合、又は、電子音楽配信によりオーディオデータトランスポートストリームが配信されてくる場合にこれらを受信することができる汎用パーソナルコンピュータである。

【 0 2 0 8 】

{64-1} 記録装置のハードウェア構成

図 6 4 は、記録装置のハードウェア構成を示す図である。本図において記録装置は、フラッシュメモリカード 3 1 を接続するためのカードコネクタ 2 1 と、RAM 2 2 と、記録装置の統合制御を行うための記録制御プログラムを格納した固定ディスク装置 2 3 と、マイクから入力された音声を A/D 変換して、PCM データを得る A/D コンバータ 2 4 と、単位時間当たりの PCM データをエンコードして、ADTS ヘッダを付与することにより、AOB_FRAME を得る ACC エンコーダ 2 5 と、各 AOB_FRAME 毎の暗号鍵を用いて、AOB_FRAME を暗号化するスクランブル部 2 6 と、電子音楽配信により SD-Audio ディレクトリイが暗号化された状態で通信回線を介して伝送されてくる場合、又は、電子音楽配信によりオーディオデータトランスポートストリームが通信回線を介して伝送されてくる場合に、オーディオデータトランスポートストリームを受信するモデム装置 2 7 と、記録装置内の統合制御を行う CPU 2 8 と、操作者からの操作を受け付けるキーボード 2 9 と、ディスプレイ 3 0 とを有する。

【 0 2 0 9 】

{64-2} 入力経路RT1～RT4

電子音楽配信によりデータ領域及び認証領域に書き込むべきSD-Audioディレクトリが暗号化された状態で通信回線を介して伝送されてくる場合、記録装置はこれらを正当に受信して時点で、フラッシュメモリカード 3 1 のデータ領域及び認証領域に書き込めばよい。しかし、SD-Audioディレクトリでは無く、電子音楽配信によりオーディオデータトランスポートストリームそのものが伝送されてくる場合、またPCMデータの状態で記録装置に入力されてくる場合、原音の状態で記録装置に入力されてくる場合、記録装置は、以下に示す3つの入力経路を経て、フラッシュメモリカード 3 1 にオーディオデータトランスポートストリームを書き込むこととなる。

【 0 2 1 0 】

本図における記録装置がフラッシュメモリカード 3 1 に、オーディオデータトランスポートストリームを格納する際、オーディオデータトランスポートストリームの入力経路には、図 6 4 に示す入力経路RT1、入力経路RT2、入力経路RT3、入力経路RT4がある。

{64-3} 入力経路RT1

入力経路RT1は、電子音楽配信によりSD-Audioディレクトリが暗号化された状態で通信回線を介して伝送されてくる場合、又は、オーディオデータトランスポートストリームが通信回線を介して伝送されてくる場合の入力経路であり、この場合、トランスポートストリームに含まれるA0B_FRAMEは、一個のA0Bに属するもの毎に異なる暗号鍵を用いて暗号化されている。暗号化済みのトランスポートストリームに対しては、再度の暗号化、符号化の必要は存在しないので、SD-Audioディレクトリ又はオーディオデータトランスポートストリームは、暗号化された状態で、RAM 2 2 に格納される。

【 0 2 1 1 】

{64-4} 入力経路RT2

入力経路RT2は、マイクから音声が入力されてくる場合の入力経路である。この場合、A/Dコンバータ 2 4 にマイクから入力された音声をA/D変換を行わせて、

PCMデータを得る。そしてPCMデータのエンコードをAACエンコーダ 2 5に行わせて、ADTSヘッダを付与することにより、AOB_FRAMEを得る。その後、スクランブル部 2 6に各AOBファイル毎の暗号鍵を用いて、AOB_FRAMEを暗号化を行わせることにより、暗号化がなされたオーディオデータを得る。その後、オーディオデータをRAM 2 2に格納する。

【 0 2 1 2 】

{64-5} 入力経路RT3

入力経路RT3は、CDから読み出されたPCMデータが装置に入力されてくる場合の入力経路である。PCMデータの状態で入力されてくるので、当該PCMデータは、そのままAACエンコーダ 2 5に入力される。そのように入力されたPCMデータのエンコードをAACエンコーダ 2 5に行わせて、ADTSヘッダを付与することにより、AOB_FRAMEを得る。その後、スクランブル部 2 6に各AOB_FRAME毎の暗号鍵を用いて、AOB_FRAMEを暗号化を行わせることにより、暗号化がなされたオーディオデータを得る。その後、オーディオデータはRAM 2 2に格納される。

【 0 2 1 3 】

{64-6} 入力経路RT4

入力経路RT4は、3つの入力経路RT1,RT2,RT3において入力されたトランスポートストリームを、フラッシュメモリカード 3 1に書き込む際の入力経路である。

かかるオーディオデータの格納に伴い、TKI、Default_Playlist情報は生成される。再生装置の場合と同様、かかる記録装置の機能主体も、ROMに記録されている記録プログラムである。即ち、AOBの記録やTrackManager、PlayListManagerの記録といった本実施形態特有の処理は、固定ディスク装置に記録されている記録プログラムにより実現される。

【 0 2 1 4 】

{64-7_65} 記録処理の処理手順

以降、フローチャートを参照しながら、上記入力経路RT1,RT2,RT3,RT4において、トランスポートストリームをフラッシュメモリカード 3 1に書き込む場合の記録処理の処理手順について説明する。図 6 5は、記録処理の処理手順を示すフローチャートである。本フローチャートにおいて引用する変数としては、Frame_

Number、Data_Sizeといったものがある。Frame_Numberとは、これまでAOBファイルに記録されたAOB_FRAMEの総数を管理するための変数であり、Data_Sizeとは、これまでAOBファイルに記録されたAOB_FRAMEのデータサイズを管理するための変数である。

【 0 2 1 5 】

本フローチャートが実行されると、ステップS 2 0 0においてCPU 2 8は、DefaultPlaylist,TrackManagerを作成し、ステップS 2 0 1において、変数#zを初期化する($z \leftarrow 1$)。ステップS 2 0 2では、AOBファイル#zを作成してフラッシュメモリカード3 1におけるデータ領域に格納する。この状態で、データ領域におけるSD-Audioディレクトリのディレクトリエントリーには、AOBファイル#zのファイル名、ファイル拡張子、最初のクラスタ番号が設定されることになる。続くステップS 2 0 3において、CPU 2 8は、TKI#zを作成してTrackManagerに格納し、ステップS 2 0 4においてCPU 2 8は、DPL_TK_SRP#zを作成してDefaultPlaylist情報に格納する。以降ステップS 2 0 5において変数#yを初期化し($y \leftarrow 1$)、ステップS 2 0 6において、Frame_Number、Data_Sizeのそれぞれを初期化する($\text{Frame_Number} \leftarrow 0$ 、 $\text{Data_Size} \leftarrow 0$)。

【 0 2 1 6 】

ステップS 2 0 7においてCPU 2 8は、AOBファイル#zに書き込むべき、オーディオデータトランスポートストリームの入力終了したかを判定する。AACエンコーダ2 5により符号化され、スクランブル部2 6により暗号化されたオーディオデータトランスポートストリームが続々とRAM 2 2に格納されており、クラスタデータの書き込みを継続する必要がある場合、ステップS 2 0 7がNoとなりステップS 2 0 9に移行する。ステップS 2 0 9においてCPU 2 8は、クラスタサイズ分のAACオーディオデータがRAM 2 2に蓄積したかを判定する。RAM 2 2にクラスタデータが蓄積した場合、ステップS 2 0 9がYesとなり、ステップS 2 1 0に移行して、RAM 2 2に蓄積されたクラスタサイズのAACオーディオデータをフラッシュメモリカード3 1に書き込んだ後ステップS 2 1 1に移行する。クラスタデータの蓄積が完了していない場合、ステップS 2 1 0をスキップしてステップS 2 1 1に移行する。ステップS 2 1 1においてCPU 2 8は、Frame_Numberを

インクリメントし($\text{Frame_Number} \leftarrow \text{Frame_Number} + 1$)、 Data_Size をその AOB_FRAME のデータサイズだけインクリメントする。かかる更新を行った後、ステップ S 2 1 2 において Frame_Number が、『 FNs_Middle_TMSRTE 』として定めているフレーム数に達したか否かを判定する。ここで『 FNs_Middle_TMSRTE 』は、オーディオデータトランスポートストリームが符号化された際のサンプリング周波数に応じた値となるので、 Frame_Number が、『 FNs_Middle_TMSRTE 』に達した場合は、ステップ S 2 1 2 はYesとなるが、そうでない場合、ステップ S 2 1 2 はNoになり、ステップ S 2 0 7 に移行する。以降、ステップ S 2 0 7、ステップ S 2 1 2 がYesとなるまで、ステップ S 2 0 7 ～ステップ S 2 1 2 は繰り返し行われる。

【 0 2 1 7 】

Frame_Number が『 FNs_Middle_TMSRTE 』に達してステップ S 2 1 2 がYesとなった場合、ステップ S 2 1 2 からステップ S 2 1 3 に移行し、 Data_Size を $\text{AOB_ELEMENT}\#y$ についての $\text{TMSRT_entry}\#y$ として $\text{TKI}\#z$ の TKTMSRT に格納してステップ S 2 1 4 において $\#y$ をインクリメントした後($y \leftarrow y + 1$)、ステップ S 2 1 5 において変数 y が252に達したか否かを判定する。ここで252という値は、一個の AOB に格納できる AOB_ELEMENT の総数を示す値であり、変数 y が252に達しない場合、ステップ S 2 1 6 に移行する。ステップ S 2 1 6 では、無音状態が所定時間以上継続しており、オーディオデータが曲間の境界に達したか否かを判定する。無音状態が存在しない場合、ステップ S 2 0 6 ～ステップ S 2 1 5 の処理を繰り返し行う。変数 y が252に達した場合、又は、無音状態が所定時間以上継続した場合、ステップ S 2 1 5、ステップ S 2 1 6 の何れか一方がYesとなり、ステップ S 2 1 7 に移行して、変数 $\#z$ をインクリメントする($z \leftarrow z + 1$)。その後、インクリメントされた $\#z$ について、ステップ S 2 0 2 ～ステップ S 2 1 6 の処理を繰り返し行う。かかる繰り返しにより、複数の AOB_ELEMENT を含む AOB が次々とフラッシュメモリカード 3 1 に書き込まれてゆく。

【 0 2 1 8 】

ここで、AACエンコーダ 2 5、スクランブル部 2 6、モデム装置 2 7 からのオーディオデータトランスポートストリームの伝送が終了した場合、 $\text{AOB_ファイル}\#z$ に書き込むべき、オーディオデータトランスポートストリームの入力終了し

たことになるので、ステップ S 2 0 7 が Yes となり、ステップ S 2 0 8 に移行する。ステップ S 2 0 8 において CPU 2 8 は、Data_Size を AOB_ELEMENT#y についての TMSRT_entry#y として TKI#z の TKTMSRT に格納し、RAM22 に蓄積されたオーディオデータを AOB#z に対応する AOB ファイルに格納した後、本フローチャートの処理を終了する。

【 0 2 1 9 】

以上の処理により、暗号化されたオーディオデータトランスポートストリームは、フラッシュメモリカード 3 1 に格納されたことになるが、この暗号化を解除するための暗号鍵は、以下の処理により、認証領域に格納される。

入力経路 RT2, RT3 の場合は、一個の AOB の符号化が開始される度に、CPU 2 8 は異なる暗号鍵を生成して、スクランブル部 2 6 に設定し、その暗号鍵でスクランブル部 2 6 に暗号化を行わせると共に、それらの暗号鍵を認証領域に存在する暗号鍵格納ファイルの FileKey Entry 以降に格納する。

【 0 2 2 0 】

一方入力経路 RT1 の場合は、AOB ファイル、TKMG を格納したファイル、PLMG を格納したファイル、AOB 毎の異なる暗号鍵を格納した暗号鍵格納ファイルは、電子音楽配信のプロバイダより、送信される。CPU 2 8 は、それらを受信して、AOB ファイル、TKMG を格納したファイル、PLMG を格納したファイルをユーザ領域に書き込み、AOB 毎の異なる暗号鍵を格納した暗号鍵格納ファイルを認証領域に書き込む。尚、TKMG を格納したファイル、PLMG を格納したファイルのそのものを電子音楽配信にて配信するのではなく、TKMG, PLMG の元となる情報を、AOB ファイル、AOB 毎の異なる暗号鍵を格納した暗号鍵格納ファイルと共に配信し、記録装置において、この TKMG, PLMG の元となる情報を加工することにより TKMG, PLMG を得て、フラッシュメモリカードに記録しても良い。

【 0 2 2 1 】

また、本実施形態においては、説明の簡略化を図るため、記録装置と再生装置とが別途独立に存在する形態を説明したが、再生装置に記録装置の機能を兼備させてもよい。また、汎用パーソナルコンピュータの一例を用いて記録装置の構成を説明したが、これ以外にも、携帯端末を記録装置として構成してもよい。また

、本実施形態ではインターネットとの接続のために記録装置は、モデム装置 2 7 を有していたが、これに替えて、ISDN回線との接続を行うためのターミナルアダプタ等を具備していてもよい。

【0 2 2 2】

以上のように本実施形態によれば、AOBを格納したファイルは、それぞれ異なる暗号鍵にて暗号化されているので、1つのファイルを暗号化に用いられた暗号鍵が解読され、暴露されたとしても、その解読によって復号できるAOBは、1つのファイルに格納されているAOBだけであり、他のファイルに格納されたAOBには何の影響も及ぼさない。暗号鍵が暴露された場合の損失を最小限に抑える事ができる。

【0 2 2 3】

なお、上記実施形態は現状において最善の効果が期待できるシステム例として説明したに過ぎない。本発明は、その要旨を逸脱しない範囲で実施変更することができる。具体的には、以下の（a）（b）（c）示すような変更実施が可能である。

（a）本実施の形態では、記録媒体を半導体メモリ（フラッシュメモリカード）として説明を行ったが、これに限られるものではなく、DVD-RAMなどの光ディスクやハードディスクなどに置きかえることができる。

【0 2 2 4】

（b）本実施の形態では、音楽データとしてAACを使用したか、これに限られるものではなく、MP3（MPEG 1 Audio Layer 3）、Dolby-AC3、DTS（Digital Theater System）などであってもよい。

（c）図 5 2～図 5 5、図 5 7、図 6 0～図 6 2、図 6 5 のフローチャートを参照して説明した手順等を実行形式プログラムにより実現し、これを記録媒体に記録して流通・販売の対象にしても良い。このような記録媒体には、ICカードや光ディスク、フロッピーディスク等があるが、これらに記録された機械語プログラムは汎用コンピュータにインストールされることにより利用に供される。この汎用コンピュータは、インストールした実行形式プログラムを逐次実行して、本実施形態に示した再生装置、記録装置の機能を実現するのである。

【0 2 2 5】

【発明の効果】

以上のように本発明に係る半導体メモリカードは、オーディオストリームを分割することにより得られた複数のオーディオブロックであって、各々が互いに異なる暗号鍵にて暗号化されているものを収録しているファイルからなるファイル群と、前記各ファイルに収録されているオーディオブロックについての諸特性を示す管理情報をテーブル形式に配してなる管理情報テーブルと、前記各オーディオブロックをどのような順序で再生させるかを示すプレイリスト情報とを格納している。オーディオブロックを格納したファイルは、それぞれ異なる暗号鍵にて暗号化されているので、1つのファイルを暗号化に用いられた暗号鍵が解読され、暴露されたとしても、その解読によって復号できるオーディオブロックは、1つのファイルに格納されているオーディオブロックだけであり、他のファイルに格納されたオーディオブロックには何の影響も及ばないので、暗号鍵が暴露された場合の損失を最小限に抑える事ができる。

【0 2 2 6】

曲若しくは章に相当する複数のオーディオブロックをどのような順序で再生させるかは、プレイリスト情報により定められるので、プレイリスト情報の設定により、ファイルの暗号化を解除することなく、複数のオーディオブロックの再生順序を自由に変化させることができる。

上述した半導体メモリカード(1)において、前記オーディオストリームは、複数のコンテンツを含むものであり、複数のコンテンツのうち、その再生が所定の上限時間以内に完遂するものは、1つのオーディオブロックとして1つのファイルに収録されていて、再生が所定の上限時間以内に完遂しない残りのコンテンツは、複数のオーディオブロックに分割され、同数のファイルに収録されており、前記管理情報テーブルに格納されている各ファイルについての管理情報は、ファイルに収録されているオーディオブロックが、コンテンツそのものであるか、コンテンツの一部であってその先頭部分であるか、中間部分であるか、終端部分であるかを示す第1属性情報を含んでもよい(2)。

【0 2 2 7】

半導体メモリカードをこのように構成した場合、更なる効果を奏することができる。即ち、暗号化された複数のオーディオブロックをどのような曲として管理するかは、管理情報における属性情報に示されているので、例えば2つのファイルに収録されているオーディオブロックが、管理情報において2つの独立した曲として管理されている場合、これら2つのオーディオブロックにおける管理情報の属性情報を、曲の先頭部分と、曲の終端部分とに変更することにより、2つのオーディオブロックを1つの曲に統合することができる。属性情報変更により曲の統合編集が行なえるので、ファイルの暗号化をむやみに解除することなく、統合編集を高速に行うことができる。

【0228】

上述した半導体メモリカード(1)において、前記所定の上限時間は、間欠的な再生時における一スキップ分の時間を m 倍した時間であり(m は、2以上の整数)、前記管理情報は更に、オーディオブロックの内部を前記間欠的に再生する際、スキップ先となるアドレスを示す m 個以下のエン트리情報を含んでいてもよい(6)。

【0229】

オーディオストリームが音楽アルバムであり、その中に演奏時間が極端に長いものが存在する場合、演奏時間が長い曲は、エン트리情報の総数が m 個以下となるように分割されて、複数のファイルに格納されることとなる。かかる分割により、エン트리情報の総数を抑制すれば、これらを含む管理情報のデータサイズも併せて抑制することができる。

【0230】

管理情報のデータサイズ抑制により、再生装置は、以下のように管理情報を読み出し利用することができる。即ち、あるファイルが読み出されて、それに含まれるオーディオブロックの再生が開始されると、それに対応する管理情報を読み出して、メモリに格納する。以降、当該オーディオブロックの再生が継続している期間において、この管理情報をメモリに常駐させておく。当該オーディオブロックの再生が終われば、これに後続するオーディオブロックが読み出されて、その再生が開始されると、それに対応する管理情報を読み出して、それまでメモリ

上に格納されていた管理情報を、新たに読み出された管理情報を用いて上書きする。

【0231】

オーディオブロックの再生が継続している期間においてのみ、そのオーディオブロックに対応する管理情報をメモリに常駐するという処理を繰り返し行なえば、再生装置におけるメモリの実装量が小規模であっても、順方向サーチ再生、逆方向サーチ再生といった特殊再生を行うことができる。

上述した半導体メモリカード(1)において、前記ブロック情報テーブルは更に、複数のファイルのそれぞれにおいて、ファイル先頭から、それに収録されているオーディオブロックのオーディオブロックの先頭まで無効部分のデータ長を示すデータオフセット情報、オーディオブロックの末尾からそれを収録しているファイルの末尾までの間に無効領域が存在する場合に、オーディオブロックの実効データ長を示す実効データ長情報を含んでいてもよい(9)。

【0232】

オーディオフレームの前方に無効部分が存在する場合に、その無効部分のデータ長がどれだけであるか、複数オーディオエレメントにおける先頭のものから、終端のものまでの実行データ長がどれだけであるかは、ブロック情報により設定されるので、格納されている音楽が、エアチェックして録音された音楽であり、その音楽のイントロの部分にディスクジョッキーの音声が入っている場合、ブロック情報におけるデータオフセットを設定することにより、この不要音声をオーディオブロックから除外して再生させないようにすることができる。ブロック情報の設定により、不要音声を再生させないように設定することが可能なので、オーディオブロックの一部を削除するという編集がファイルの暗号化を解除することなく、高速に行うことができる。

【図面の簡単な説明】

【図1】

フラッシュメモリカード31を上面から見た場合の形状を示す図である。

【図2】

フラッシュメモリカード31をその下面から見た場合の形状を示す図である。

【図 3】

本実施形態に係るフラッシュメモリカード 3 1 の階層構造を示す図である。

【図 4】

(a) フラッシュメモリカード 3 1 の物理層に設けられた特殊領域、認証領域、ユーザ領域の構成を示す図である。

(b) ファイルシステム層における認証領域及びユーザ領域の構成を示す図である。

【図 5】

ファイルシステム層における構成の詳細を示す図である。

【図 6】

AOB001.SA1をクラスタサイズに合わせて5つに分割し、各分割部分を、クラスタ 003,004,005,00A,00Cに格納する状態を想定した図である。

【図 7】

AOB001.SA1が複数のクラスタに記録されている場合のディレクトリエントリー及びファイルアロケーションテーブルについての設定例を示す図である。

【図 8】

(a) (b) 応用層におけるこれら2つのデータを格納する場合、ファイルシステム層においてユーザ領域及び認証領域には、どのようなディレクトリが構成され、どのようなファイルが当該ディレクトリの配下に作成されるかを示す図である。

【図 9】

SD-Audioディレクトリの下にあるAOBSA1.KEYと、AOBファイルとの対応を示す図である。

【図 1 0】

AOBファイルのデータ構成を階層的に示す図である。

【図 1 1】

ISO/IEC13818-7に記述されているパラメータを表形式に示す図である。

【図 1 2】

AOB_FRAMEの構成の詳細を示す図である。

【図 1 3】

3つのAOB_FRAMEにおいて、それぞれのAOB_FRAMEにおけるオーディオデータのバイト長がどのように設定されるかを示す図である。

【図 1 4】

sampling_frequencyと、AOB_ELEMENTに含まれるAOB_FRAME数との対応を示す図である。

【図 1 5】

AOB_ELEMENTの時間長及びAOB_FRAMEの時間長の一例を示す図である。

【図 1 6】

AOBファイルに収録されている各AOB、AOB_BLOCKが連続して再生されることにより、どのような再生内容が再生されるかを示す図である。

【図 1 7】

第 1 実施形態におけるPlaylistmanager及びTrackManagerの構成を段階的に詳細化した図である。

【図 1 8】

PlayListManager及びTrackManagerのサイズを示す図である。

【図 1 9】

図 1 7 に示したTKIと、図 1 6 に示したAOBファイル及びAOBとの相互関係を示す図である。

【図 2 0】

図 1 7 に示したTKTMSRTの詳細なデータ構造を示す図である。

【図 2 1】

TKTMSRTについての一例を示す図である。

【図 2 2】

TKGIの詳細構成を示す図である。

【図 2 3】

(a) (b) BITの詳細構成を示す図である。

【図 2 4】

AOB_ELEMENT#1～#4からなるAOBが格納されているクラスタ007～クラスタ00Eを

示す図である。

【図 2 5】

A0B内の任意のA0B_ELEMENT#yにおけるA0B_FRAME#xから順方向サーチ再生を行う場合、次に再生すべきA0B_FRAME#x+1をどのように設定するかを示す図である。

【図 2 6】

(a) (b) 任意の再生開始時刻が指定された場合、その指定時刻に対応するA0B、A0B_ELEMENT、A0B_FRAMEをどのように特定するかを示す図である。

【図 2 7】

(a) (b) トラックを削除する場合を想定した図である。

【図 2 8】

(a) トラックの削除が複数回行われた後のTrackManagerを示す図である。
(b) 『Unused』のTKIが存在しており、ここに新たなTKI、A0Bファイルを書き込む場合、その書き込みがどのように行われるかを示す図である。

【図 2 9】

(a) (b) 2つのトラックを統合する場合にTKIがどのように設定されるかを示す図である。

【図 3 0】

(a) (b) 1つのトラックを2つのトラックに分割する場合を想定した図である。

【図 3 1】

(a) (b) 分割前後において、A0B003.SA1が属するSD-AudioディレクトリについてのSD-Audioディレクトリエントリーがどのように記述されているかを示す図である。

【図 3 2】

(a) A0BをA0B_ELEMENT#2の途中部分で分割する場合を想定した図である。
(b) A0B_ELEMENT#2の途中部分でA0Bが分割されて、A0B#1、A0B#2という2つのA0Bが得られた状態を示す図である。

【図 3 3】

図 3 2 に示したように AOB が分割された場合に、BIT がどのように設定されるかを示す図である。

【図 3 4】

分割の前後で BIT がどのように変化するかを更に具体的に示す図である。

【図 3 5】

分割後の TKTMSRT を示す図である。

【図 3 6】

(a) DPL_TK_SRP のフォーマットを示す図である。

(b) PL_TK_SRP のフォーマットを示す図である。

【図 3 7】

Default_Playlist 情報、TKI、AOB ファイルの相互関係を示す図である。

【図 3 8】

DefaultPlaylist、PlayList 情報の設定例を、図 3 7 と同様の表記で示した図である。

【図 3 9】

図 3 7 と同じ表記法を用いて DPL_TK_SRP と TKI との対応を示す図である。

【図 4 0】

(a) (b) 曲の順序を入れ替える場合を想定した図である。

【図 4 1】

(a) (b) 図 3 7 に示した DefaultPlaylist のうち、DPL_TK_SRP#2 及び TKI#2 を削除する場合に DefaultPlaylist、TrackManager、AOB ファイルがどのように更新されるかを示す図である。

【図 4 2】

(a) (b) 『Unused』の TKI と、DPL_TK_SRP とが存在しており、ここに新たな TKI、DPL_TK_SRP を書き込む場合、その書き込みがどのように行われるかを示す図である。

【図 4 3】

(a) (b) 曲の統合を行う場合を想定した図である。

【図 4 4】

(a) (b) 曲の分割を行う場合を想定した図である。

【図 4 5】

本実施形態に係るフラッシュメモリカード 3 1 についての携帯型の再生装置を示す図である。

【図 4 6】

プレイリストの選択が行われる際の液晶ディスプレイの表示内容の一例を示す図である。

【図 4 7】

(a) ~ (e) トラックの選択が行われる際の液晶ディスプレイの表示内容の一例を示す図である。

【図 4 8】

(a) ~ (c) ジョグダイヤルの操作例を示す図である。

【図 4 9】

再生装置の内部構成を示す図である。

【図 5 0】

ダブルバッファ 1 5 におけるデータ入出力がどのように行われるかを示す図である。

【図 5 1】

(a) (b) リングポインタを用いた巡回式の領域確保がどのように行われるかを示す図である。

【図 5 2】

AOBファイル読み出し処理の処理手順を示すフローチャートである。

【図 5 3】

AOB_FRAME出力処理の処理手順を示すフローチャートである。

【図 5 4】

AOB_FRAME出力処理の処理手順を示すフローチャートである。

【図 5 5】

AOB_FRAME出力処理の処理手順を示すフローチャートである。

【図 5 6】

(a) ~ (d) 液晶ディスプレイ 5 の時刻表示枠に表示される再生経過時刻が、変数Play_Timeの更新したがい、増加してゆく様子を示す図である。

【図 5 7】

順方向サーチ再生処理時におけるCPU 1 0 の処理手順を示すフローチャートである。

【図 5 8】

(a) ~ (d) 順方向サーチ再生時において、再生経過時刻がインクリメントされてゆく様子を示す図である。

【図 5 9】

(a) ~ (b) タイムサーチ機能が行われる場合の具体例を示す図である。

【図 6 0】

編集制御プログラムの処理手順を示すフローチャートである。

【図 6 1】

編集制御プログラムの処理手順を示すフローチャートである。

【図 6 2】

編集制御プログラムの処理手順を示すフローチャートである。

【図 6 3】

フラッシュメモリカード 3 1 の記録装置の一例を示す図である。

【図 6 4】

記録装置のハードウェア構成を示す図である。

【図 6 5】

記録処理の処理手順を示すフローチャートである。

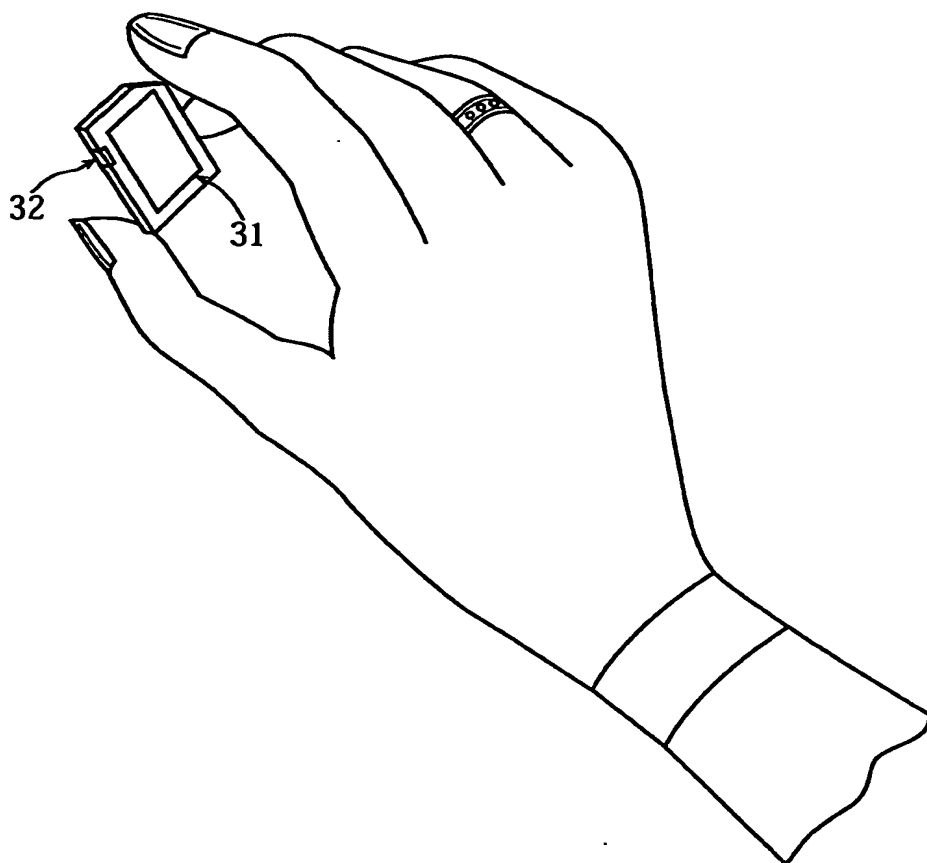
【符号の説明】

- 1 カードコネクタ
- 2 ユーザインターフェイス部
- 3 RAM
- 4 ROM
- 5 液晶ディスプレイ
- 6 LCDドライバ

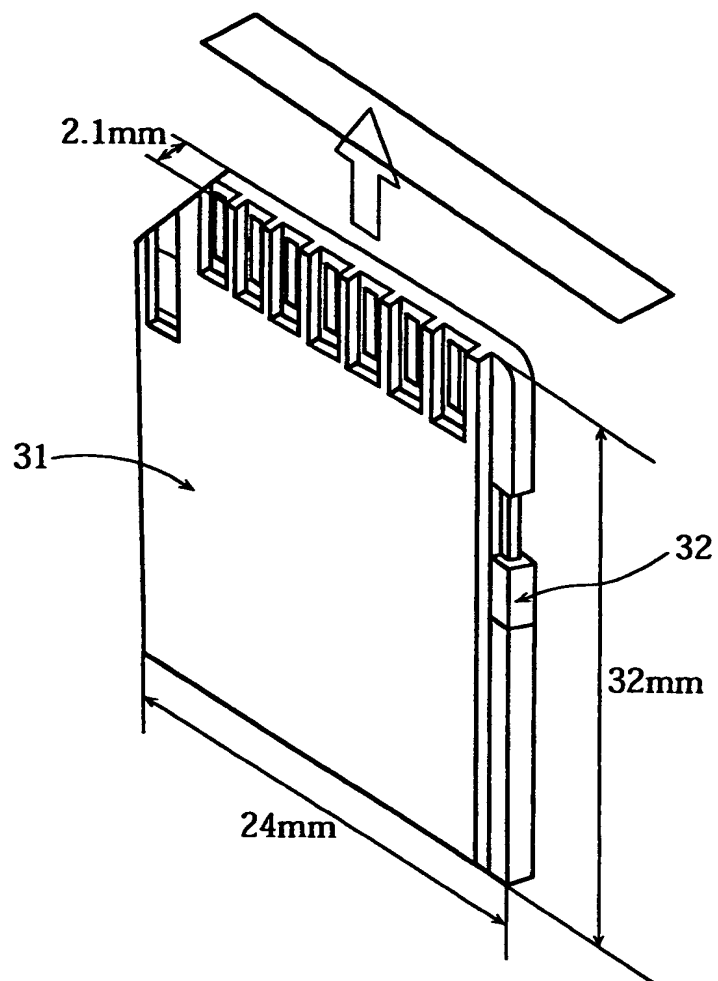
- 7 デ・スクランブラ
- 8 AACデコーダ
- 9 A/Dコンバータ
- 1 1 DPLI常駐領域
- 1 2 PLI格納領域
- 1 3 TKI格納領域
- 1 4 FileKey格納領域
- 1 5 ダブルバッファ
- 2 1 カードコネクタ
- 2 2 RAM
- 2 3 固定ディスク装置
- 2 4 コンバータ
- 2 4 A/DステップS
- 2 5 AACエンコーダ
- 2 6 スクランブル部
- 2 7 モデム装置
- 2 8 CPU
- 2 9 キーボード
- 3 0 ディスプレイ
- 3 1 フラッシュメモリカード
- 3 2 プロテクトスイッチ

【書類名】 図面

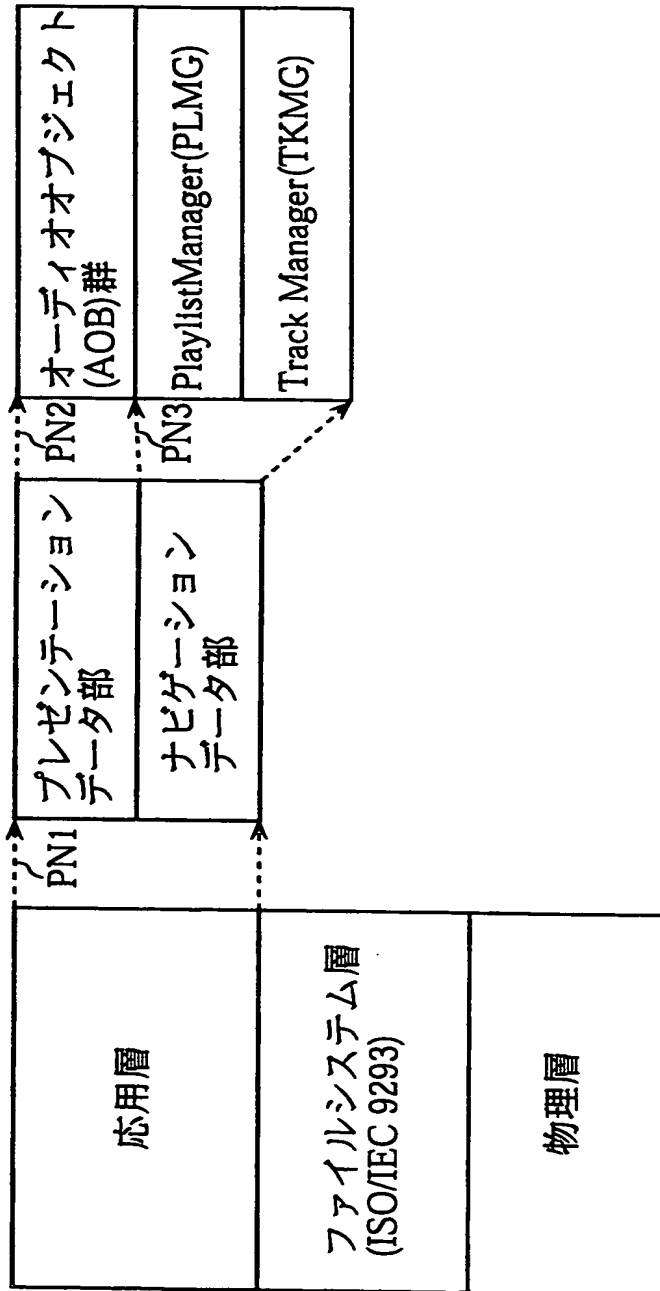
【図 1】



【図 2】

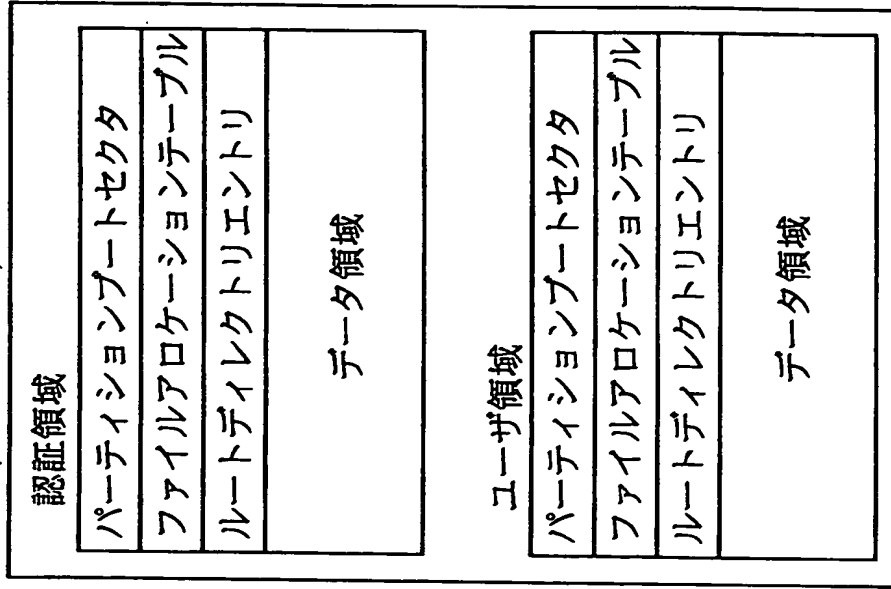


【図 3】

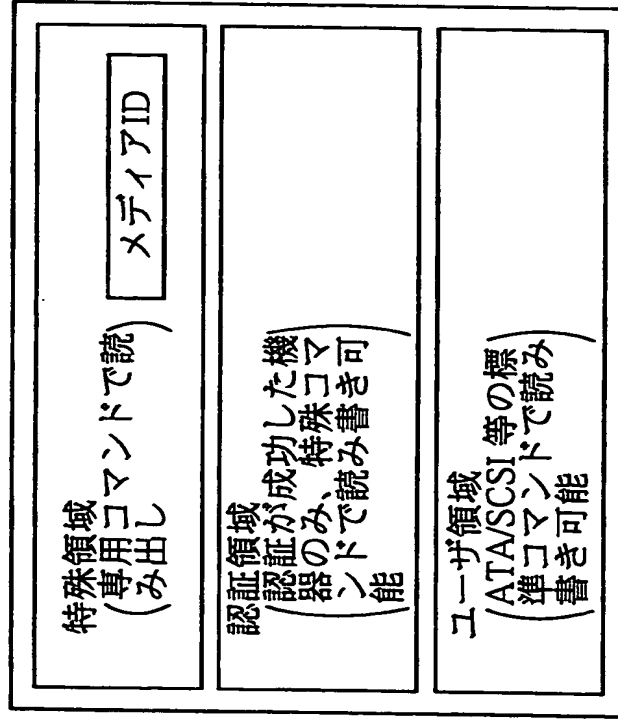


【図 4】

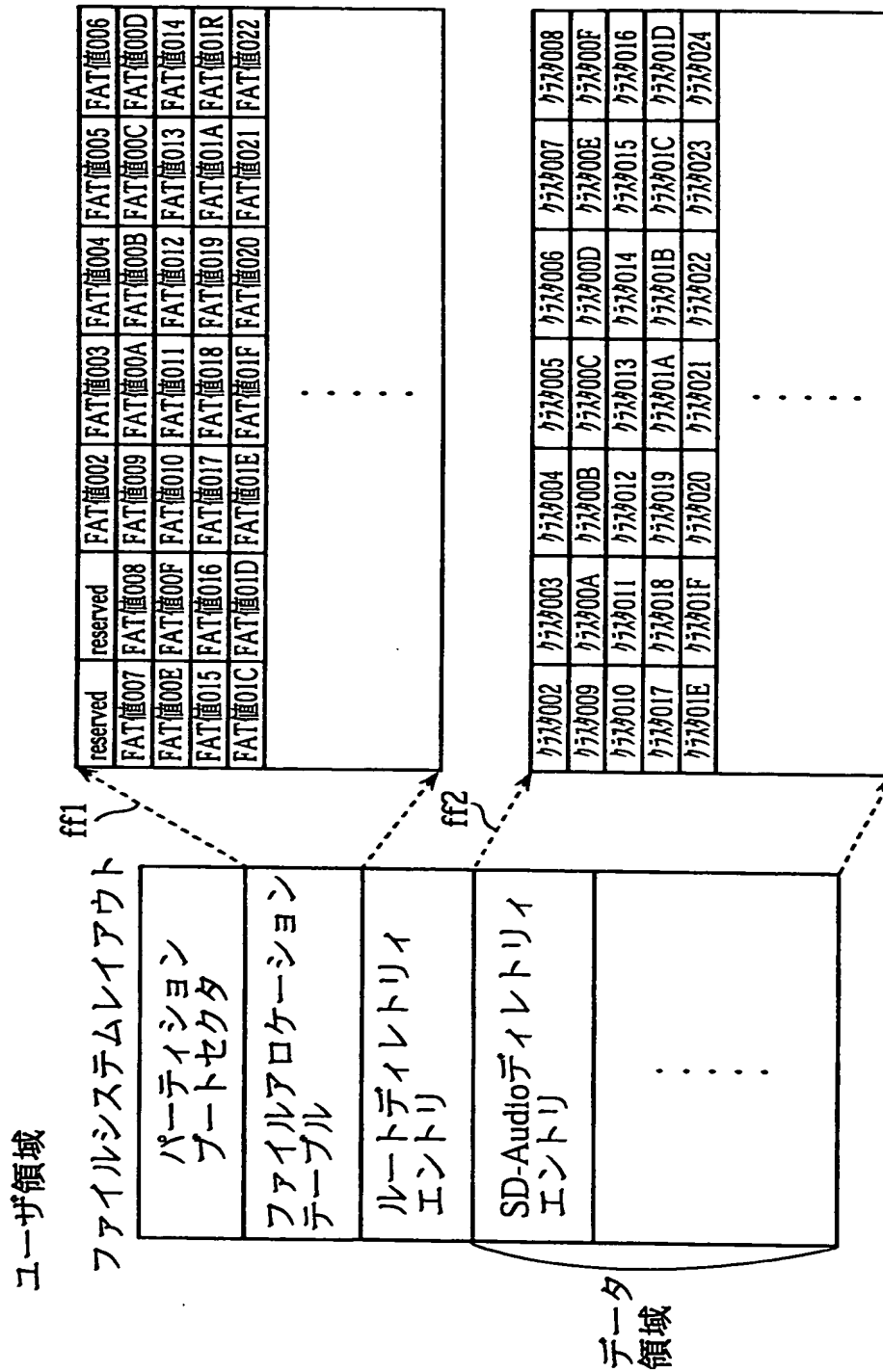
(b) ファイルシステム層
(ISO/IEC 9293)



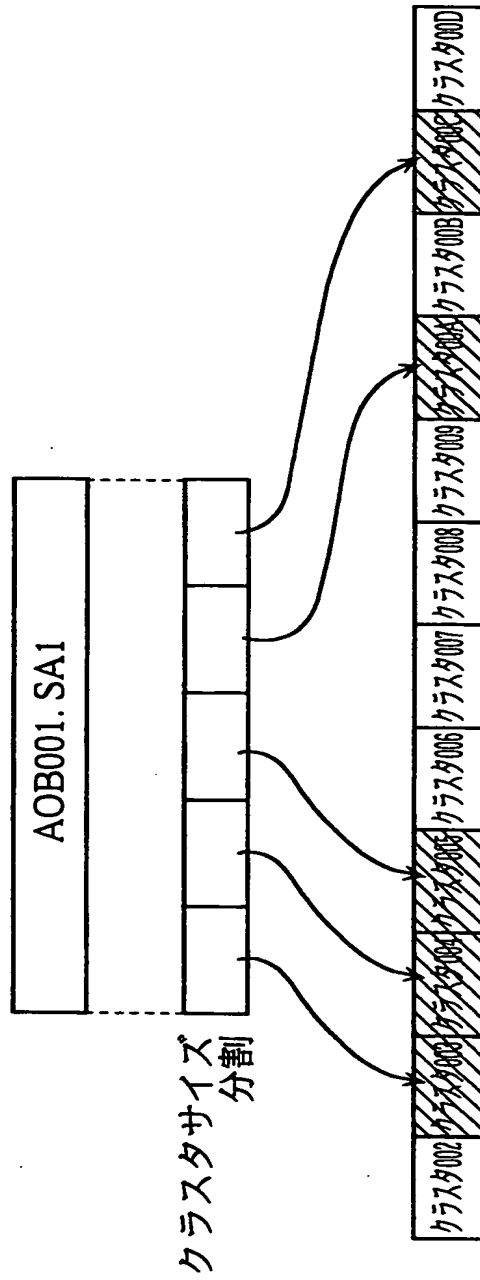
(a) 物理層



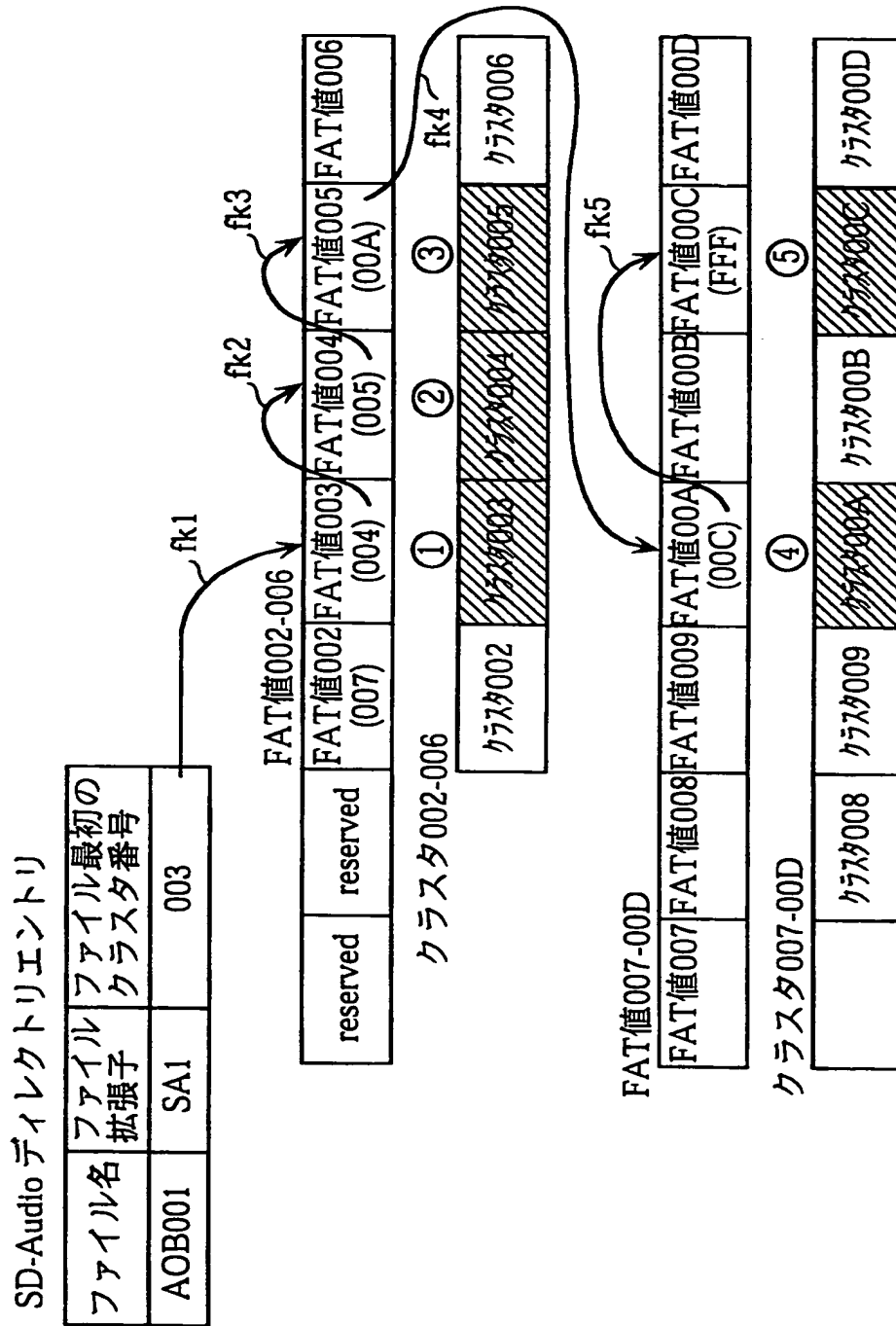
【図 5】



【図 6】

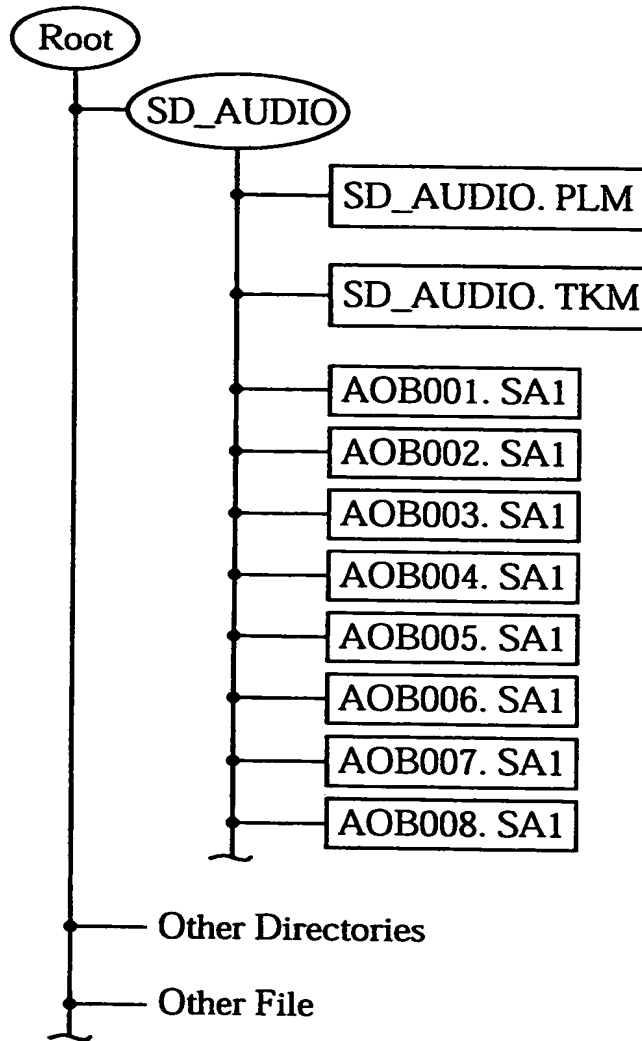


【図 7】

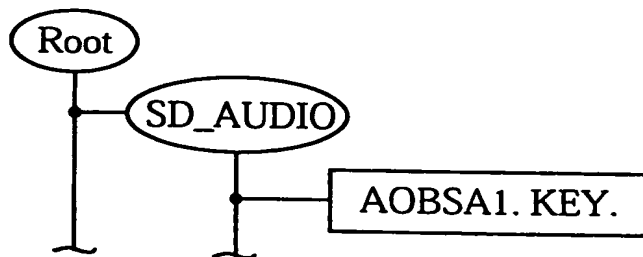


【図 8】

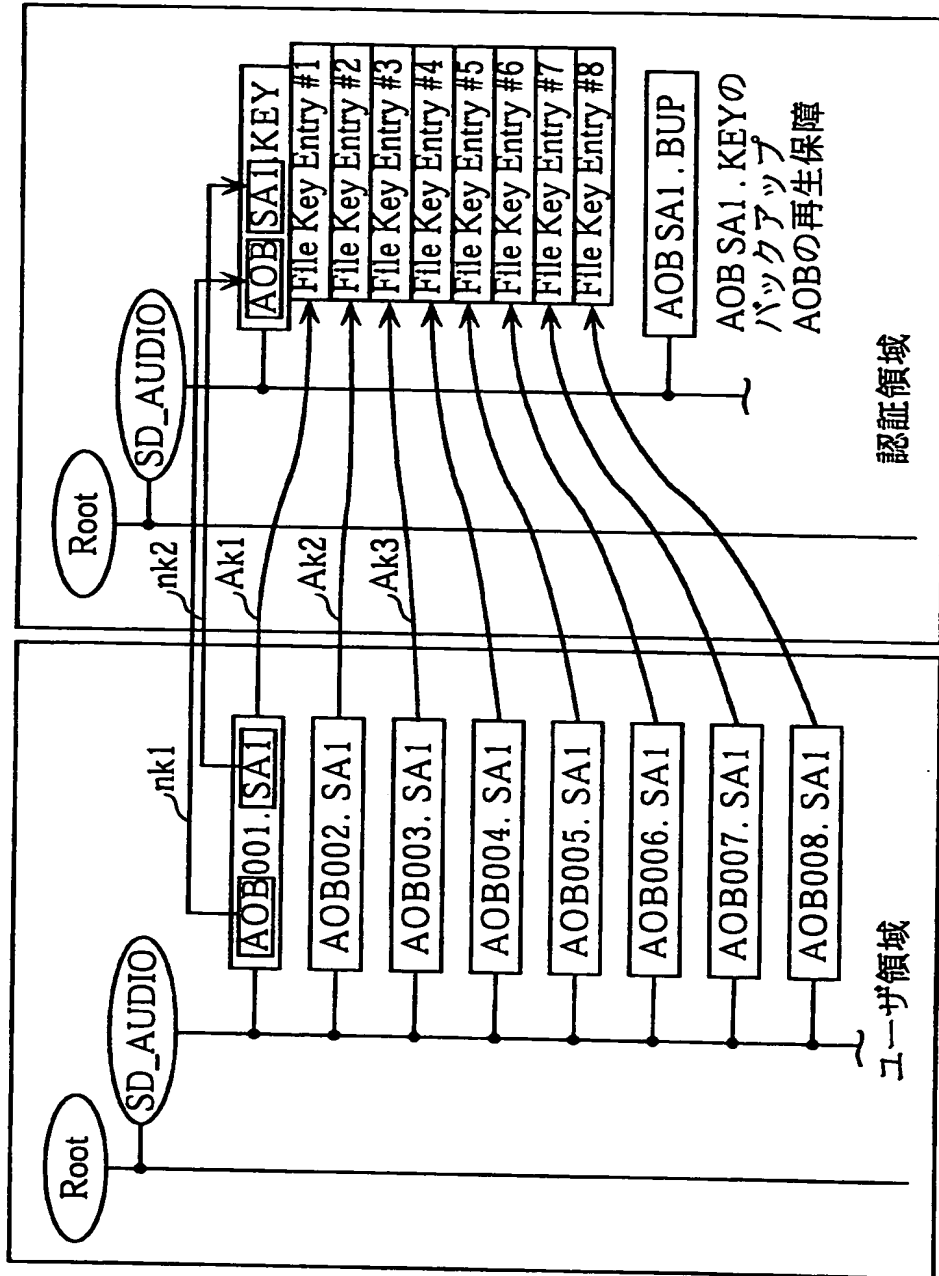
(a) ユーザ領域



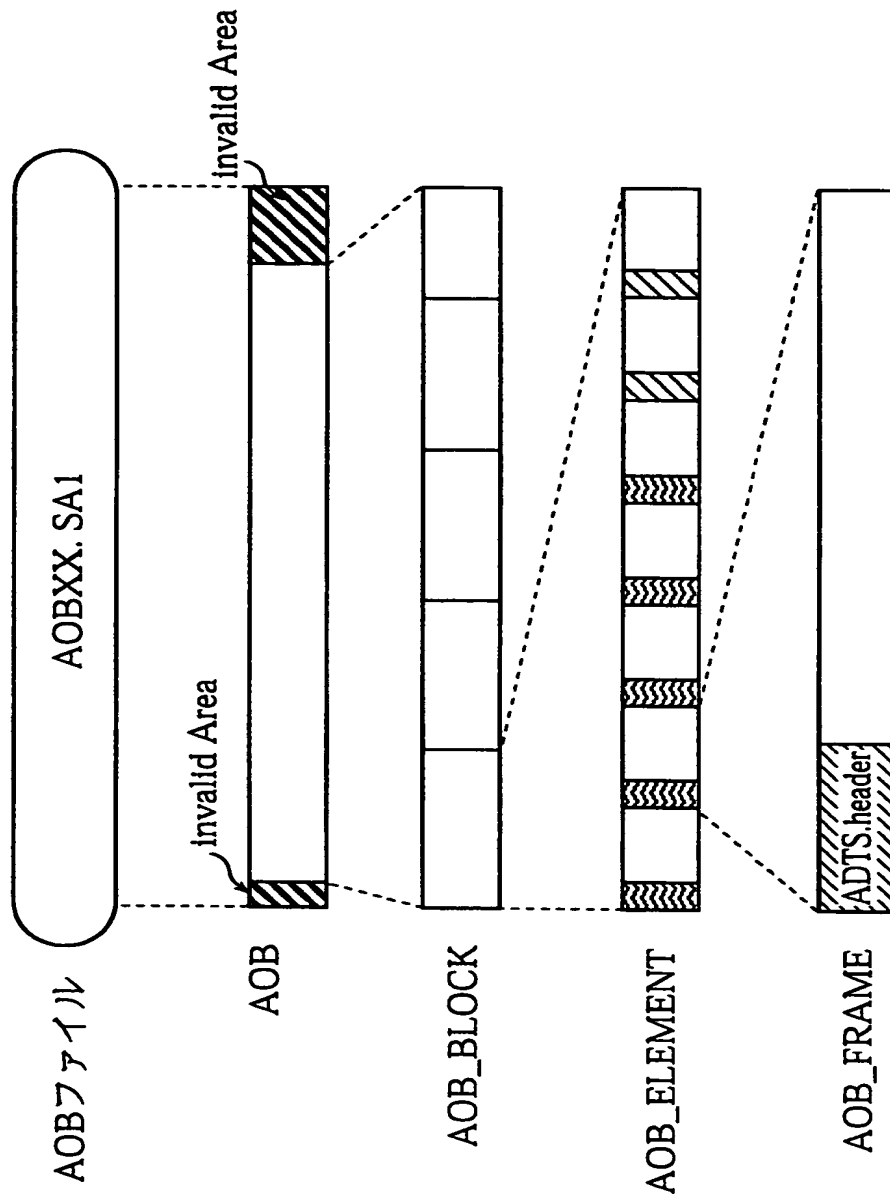
(b) 認証領域



【図 9】



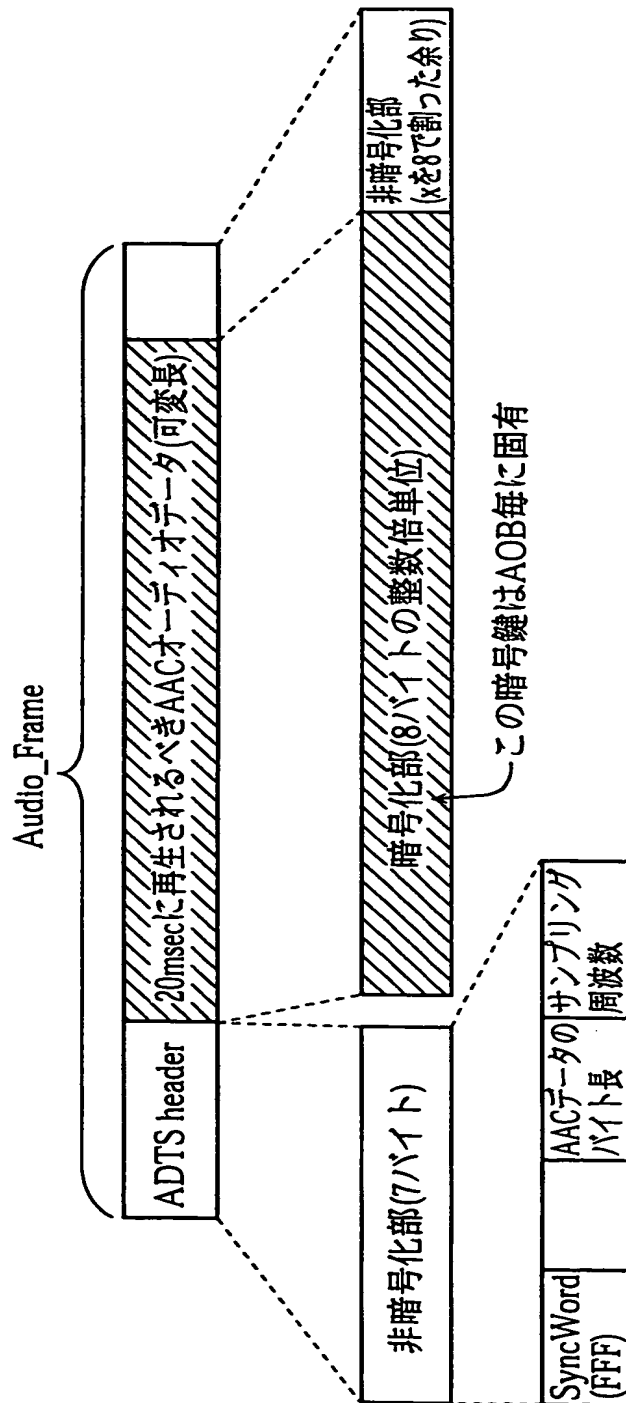
【図 1 0】



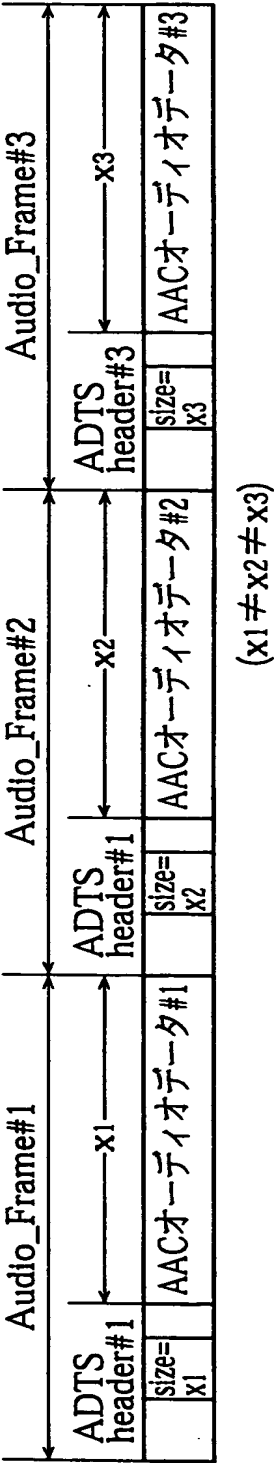
【図 1 1】

Parameter	Value	Comments
profile	01	LC profile(mandatory)
sampling_frequency_index	0011	48 kHz(mandatory)
	0100	44.1 kHz(mandatory)
	0101	32 kHz(mandatory)
	0110	24 kHz
	0111	22.05 kHz
	1000	16 kHz
	1001	12 kHz
	1010	11.025 kHz
	1011	8 kHz
	others	optional
channel_configuration	001	single_channel_element(mandatory)
	010	channel_pair_element(mandatory)
	others	optional
number_of_data_blocks_in_frame	00	1 header/1 raw_data_block(mandatory)

【図 1 2】



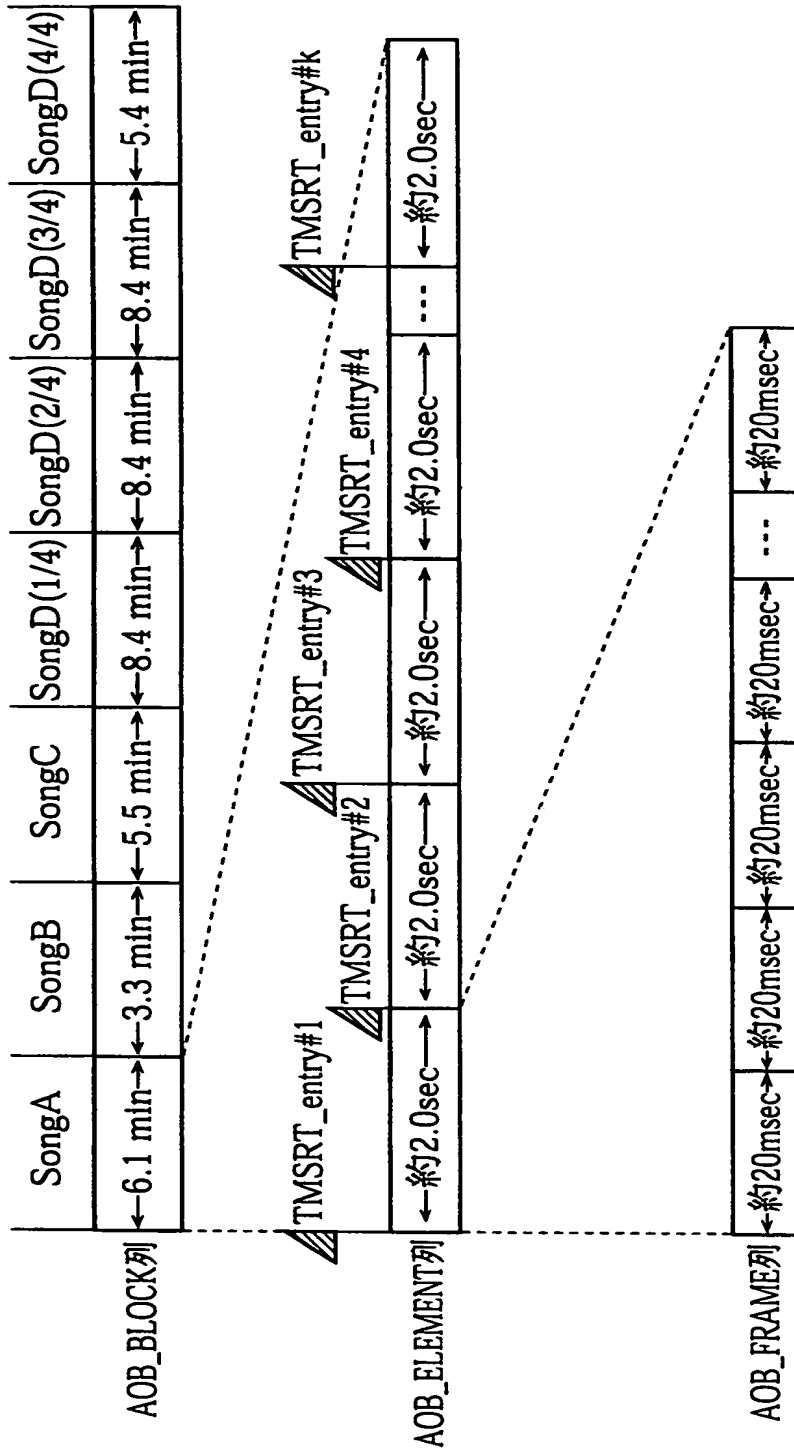
【図 1 3】



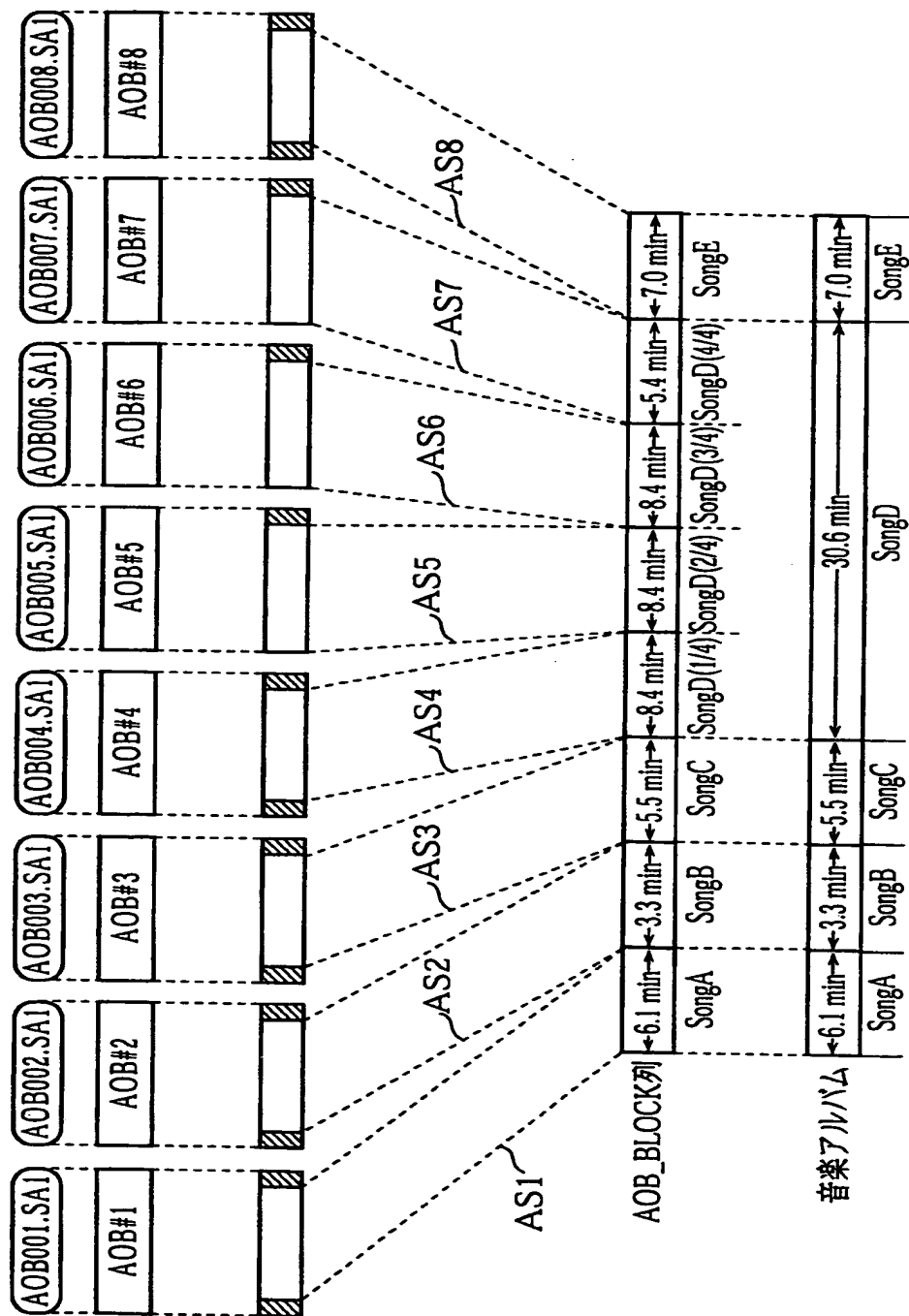
【図 1 4】

サンプリング周波数	Audio_Elementに含まれるAudio Frame数
48kHz	94
44.1kHz	87
32kHz	63
24kHz	47
22.05kHz	44
16kHz	32

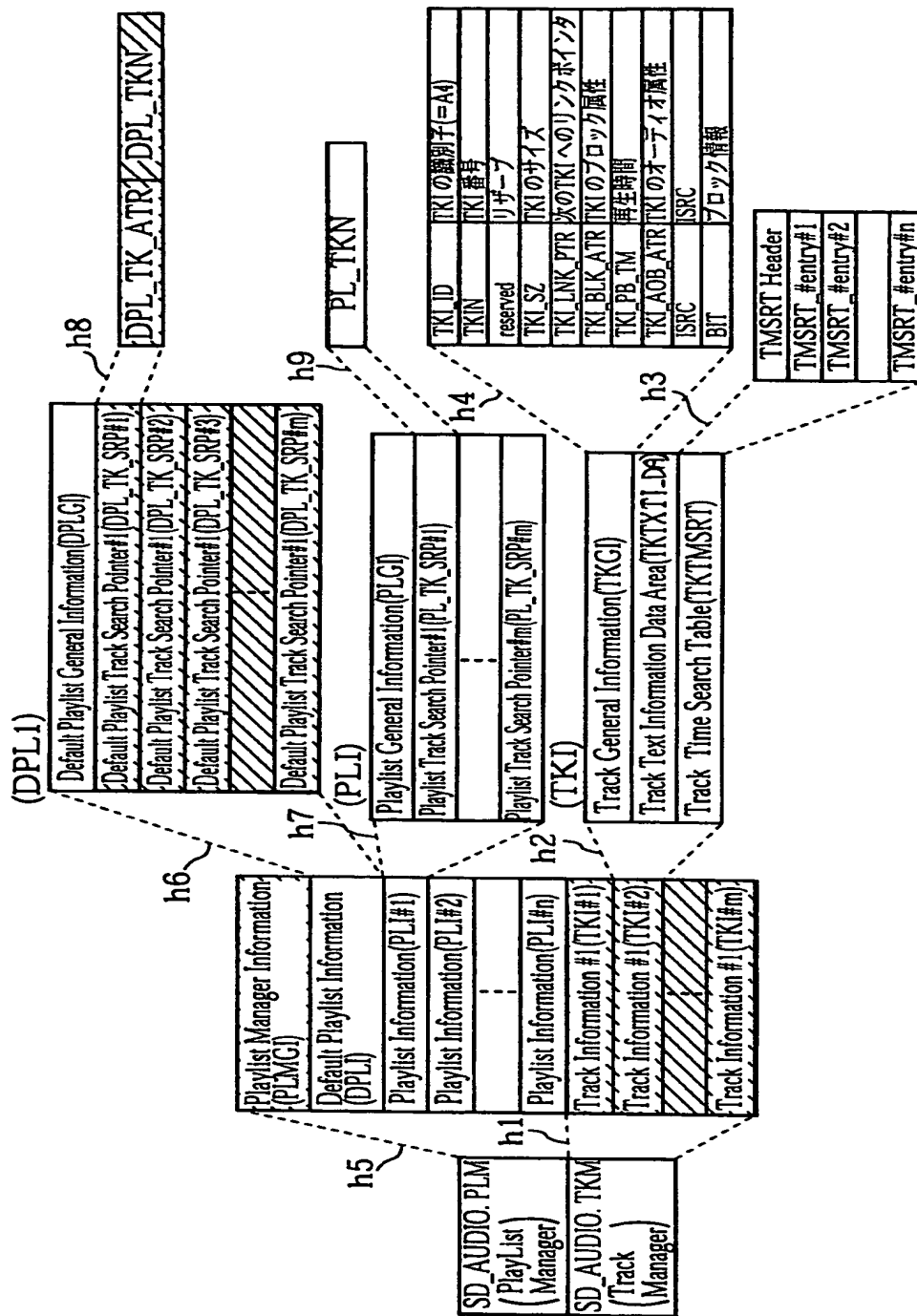
【図 1 5】



【図 1 6】



【図 17】

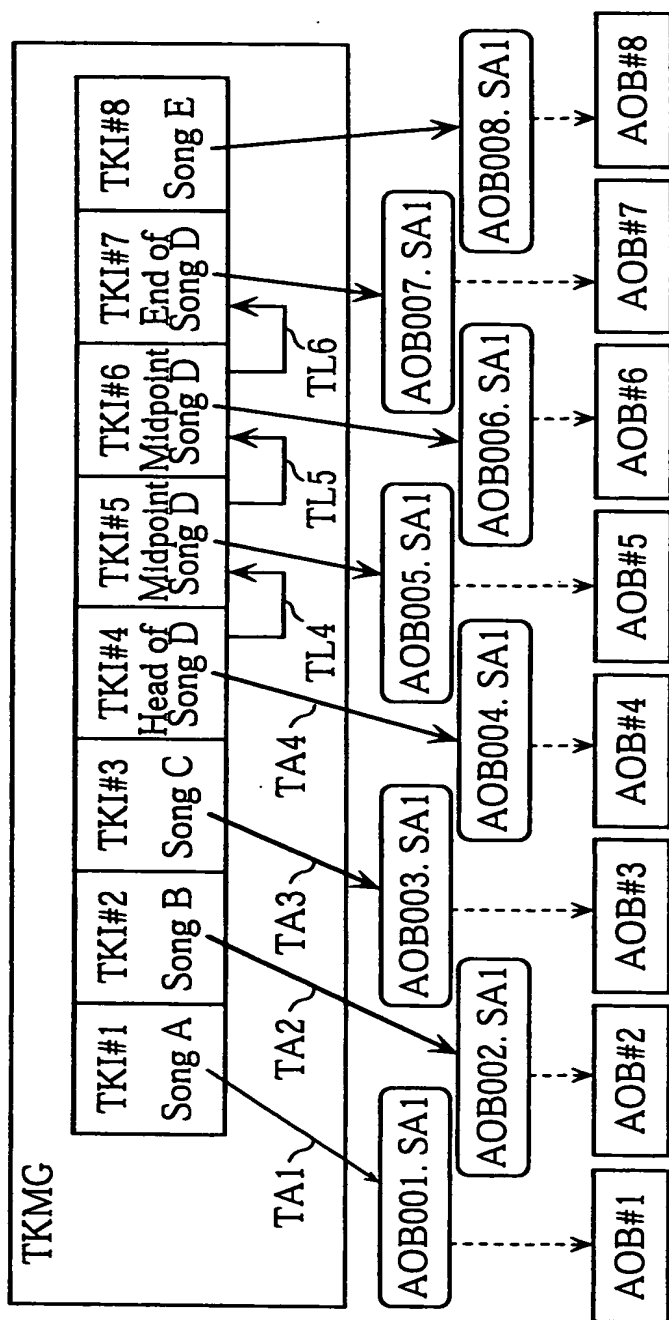


【図 1 8】

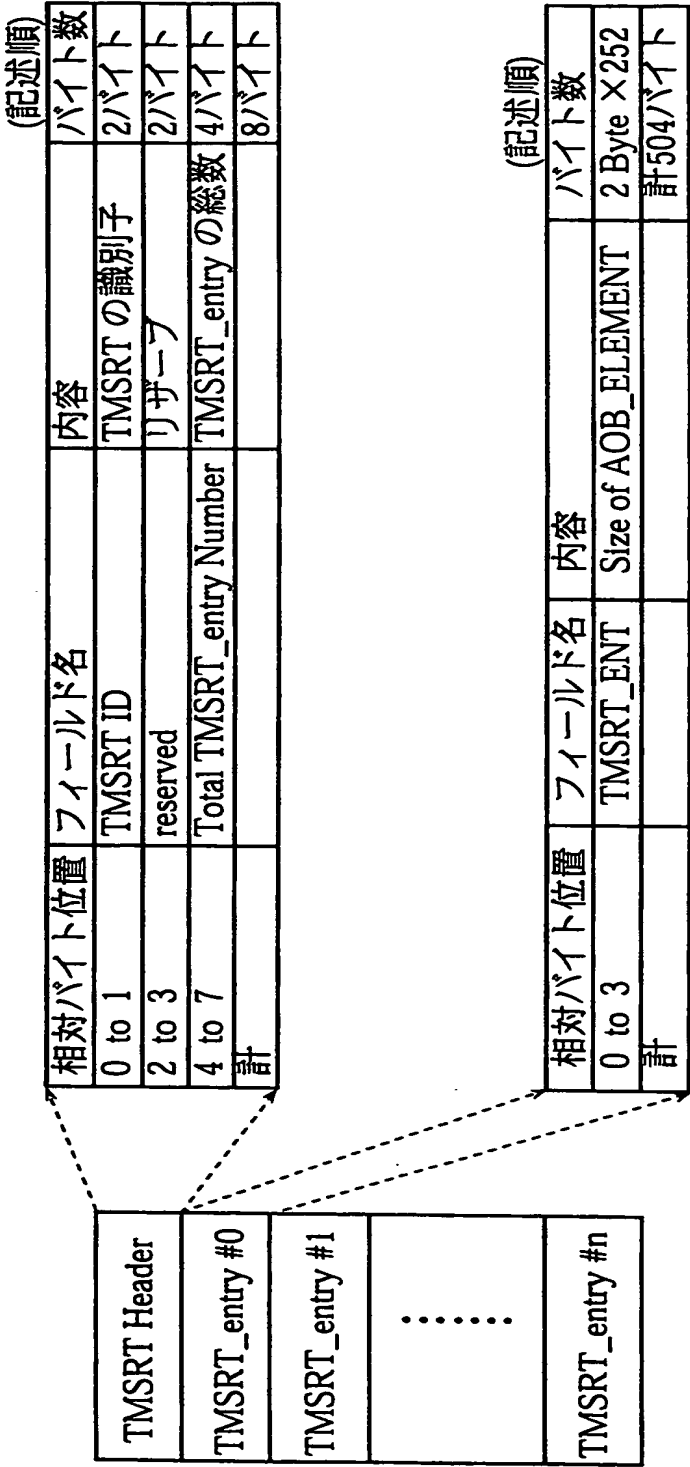
(PLMG)

Playlist Manager Information(PLMGI)	固定長 (2.5KByte)
Default Playlist Information(DPLI)	
Playlist Information(PLI#1)	固定長 (512Byte)
⋮	
Playlist Information(PLI#n) $(1 \leq n \leq 99)$	固定長 (512Byte)
Track Information #1 (TKI#1)	固定長 (1024Byte)
⋮	
Track Information #1 (TKI#m) $(1 \leq m \leq 999)$	固定長 (1024Byte)

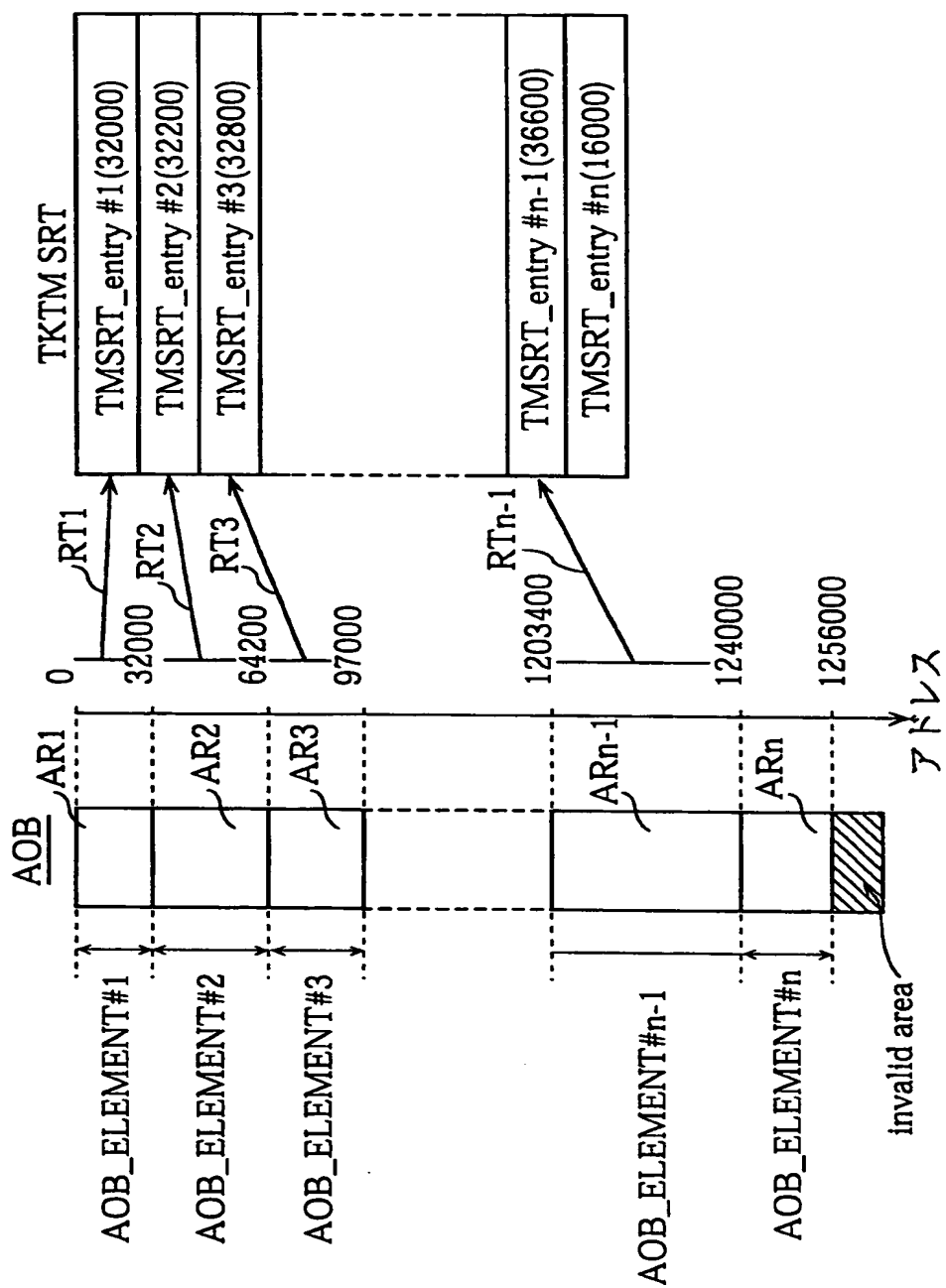
【図 1 9】



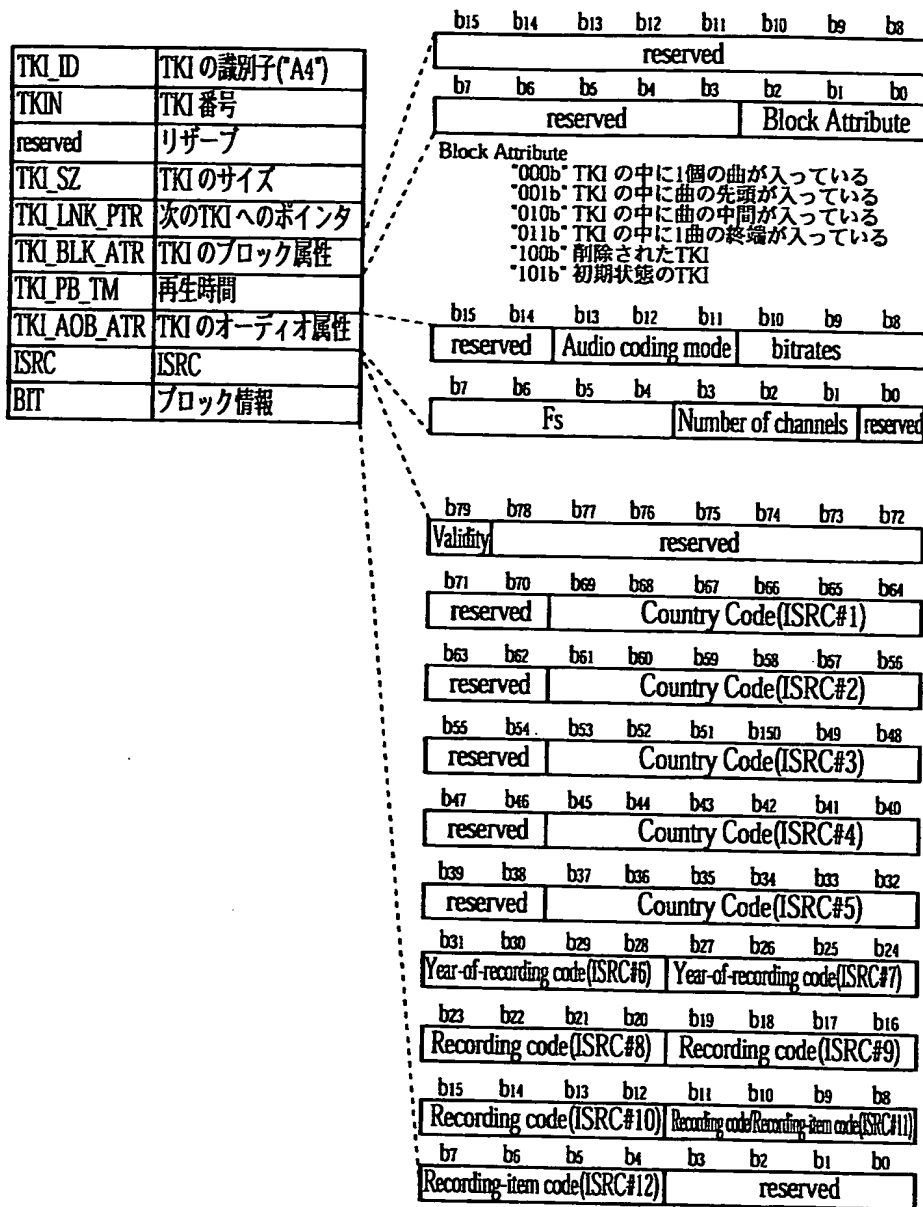
【図 2 0】



【図 2 1】



【図 2 2】



【図 2 3】

(a)

相対バイト位置	フィールド名	内容	バイト数
60 to 63	DATA_OFFSET	AOB_BLOCK の先頭アドレス	4バイト
64 to 67	SZ_DATA	AOB_BLOCK のデータ長	4バイト
68 to 71	TMSRTE_Ns	TMSRT_entry 数	4バイト
72 to 73	FNs_1st_TMSRTE	先頭AOB_ELEMENT における AOB_FRAME 数	2バイト
74 to 75	FNs_Last_TMSRTE	最終AOB_ELEMENT における AOB_FRAME 数	2バイト
76 to 77	FNs_Middle_TMSRTE	AOB_ELEMENT における AOB_FRAME 数	2バイト
計			28バイト

b15b14b13b12b11b10b9b8

reservedFNs_Middle_TMSRTE[1...8]

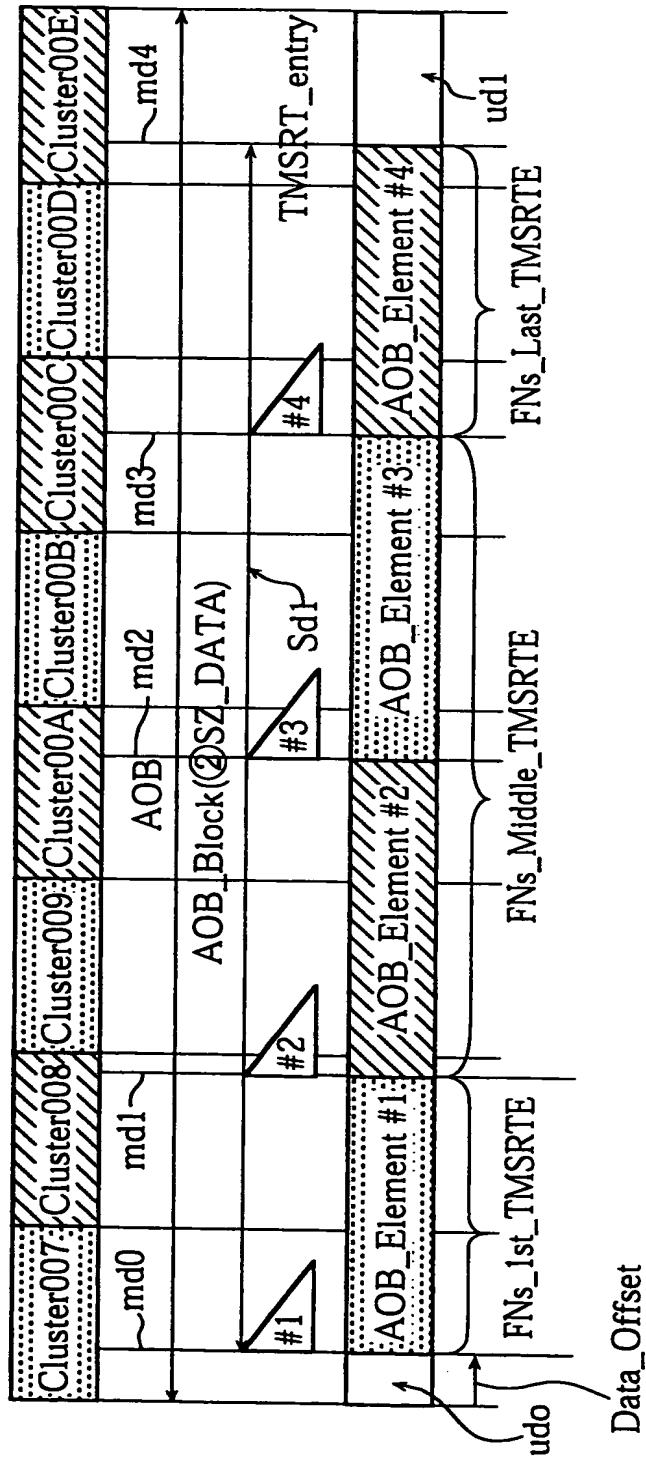
b7b6b5b4b3b2b1b0

FNs_Middle_TMSRTE[7...0]

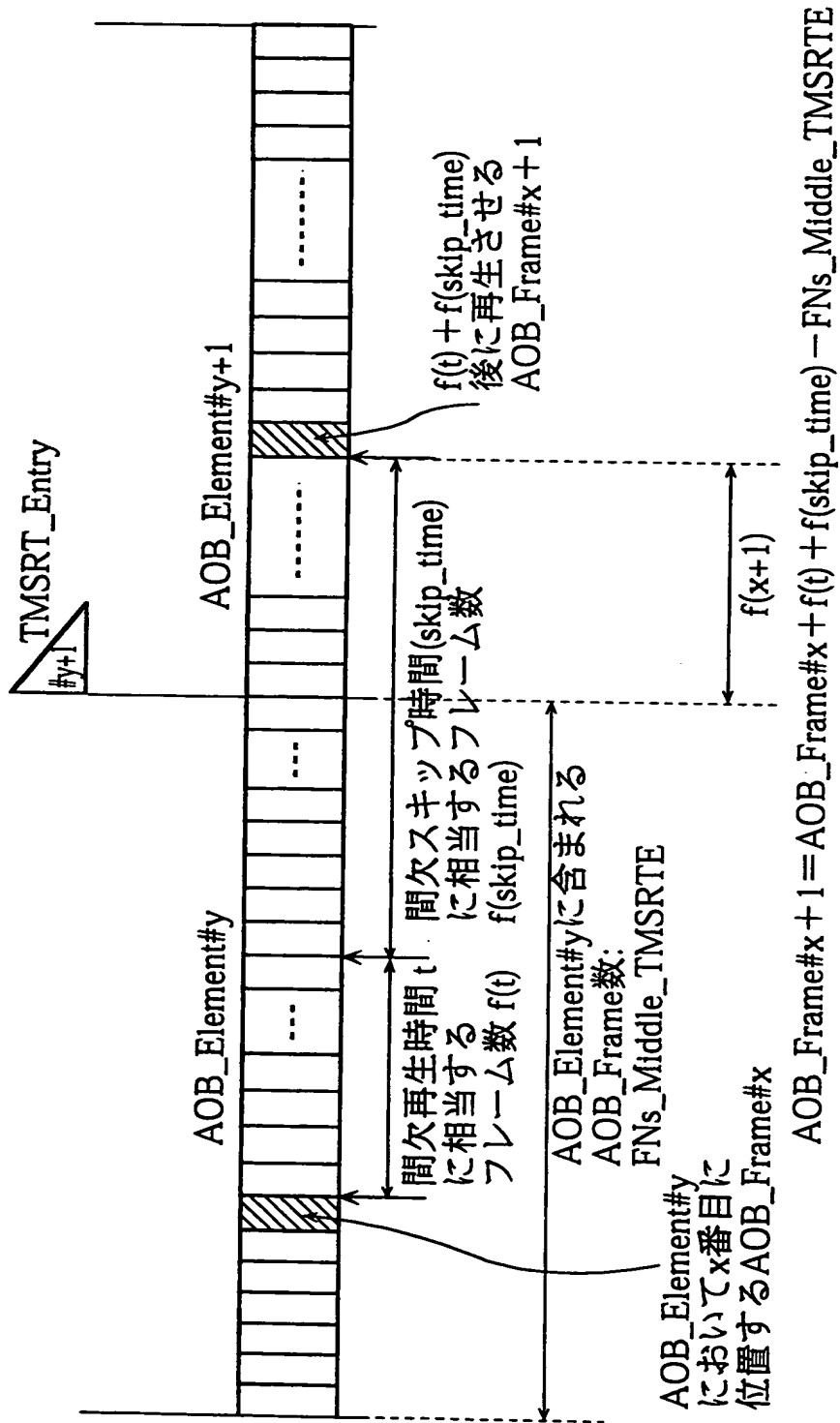
(b)

サンプリング周波数	FNs_Middle_TMSRTE
48kHz	94
44.1kHz	87
32kHz	63
24kHz	47
22.05kHz	44
16kHz	32

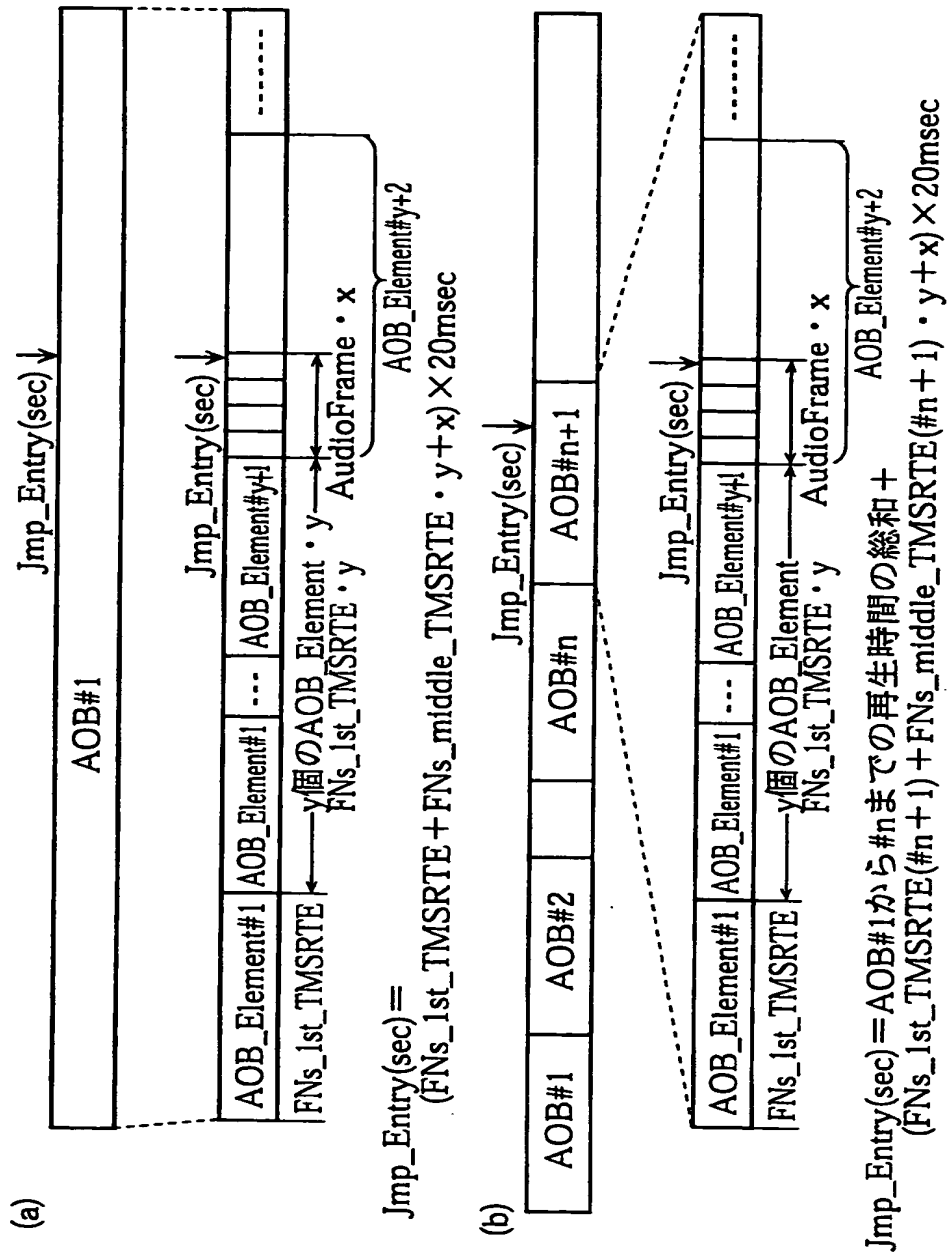
【図 2 4】



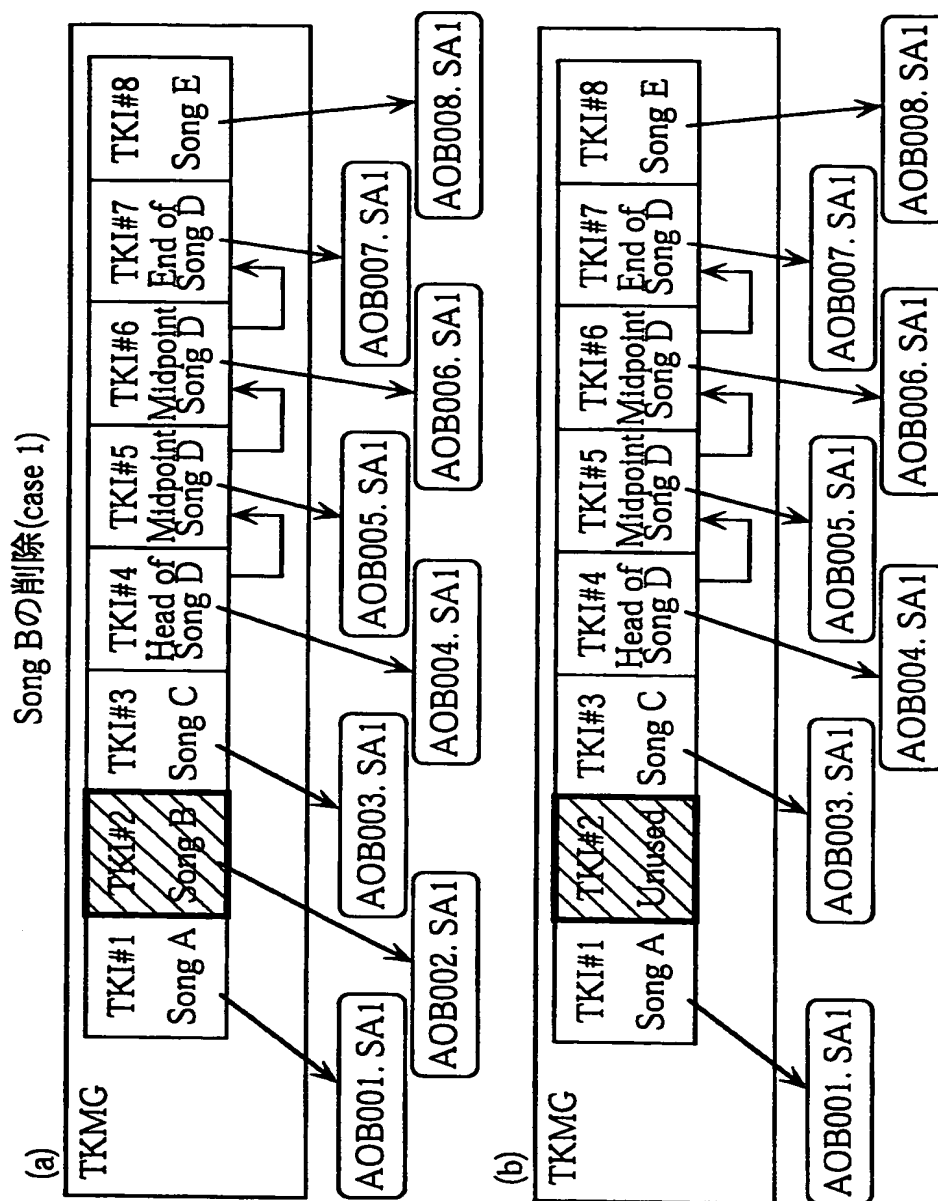
【図 2 5】



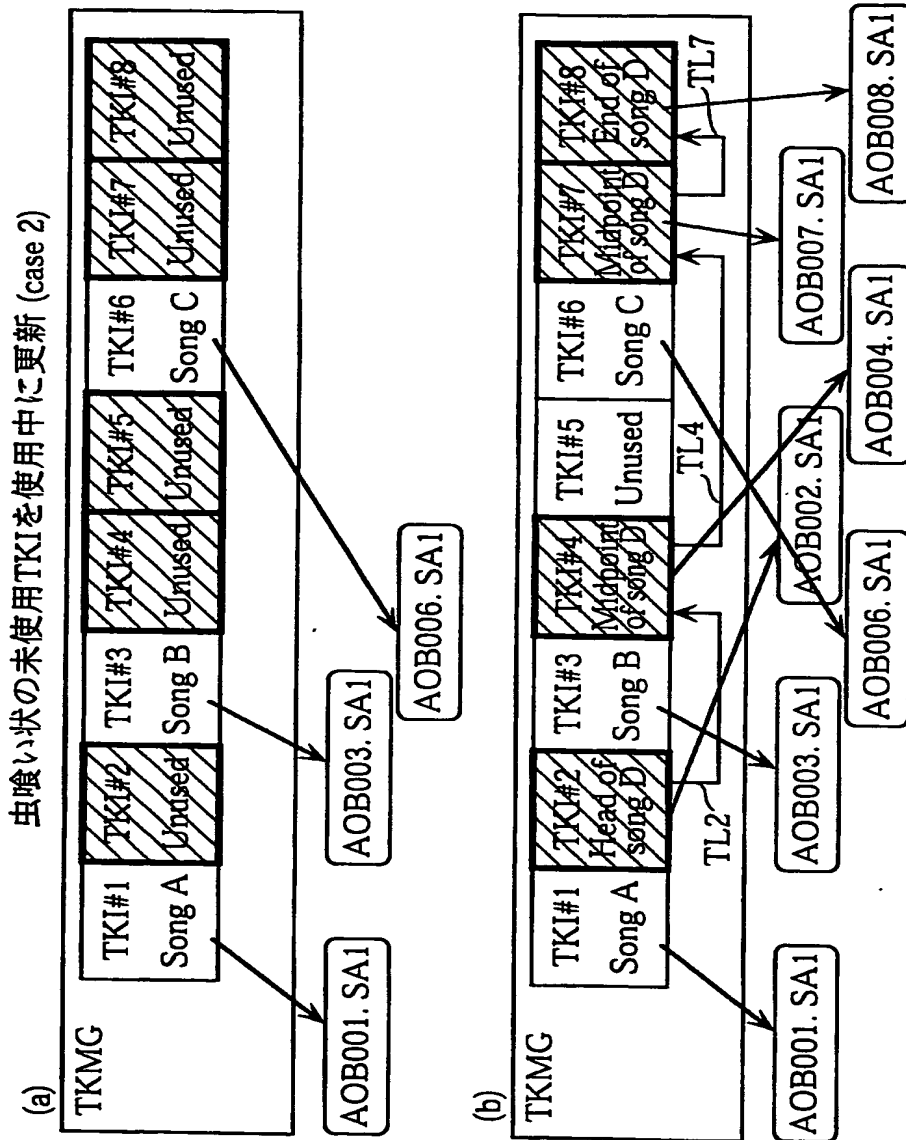
【図 2 6】



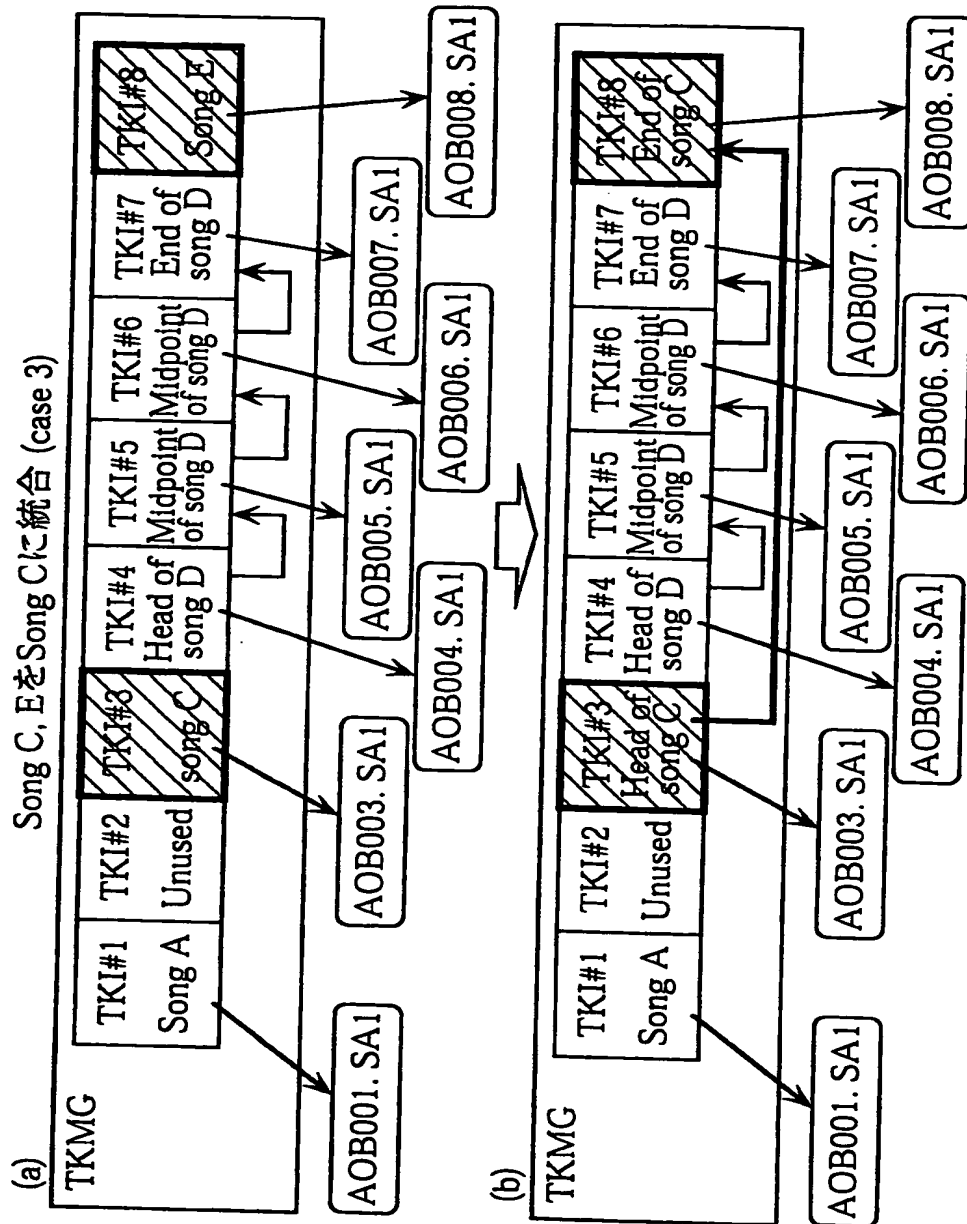
【図 2 7】



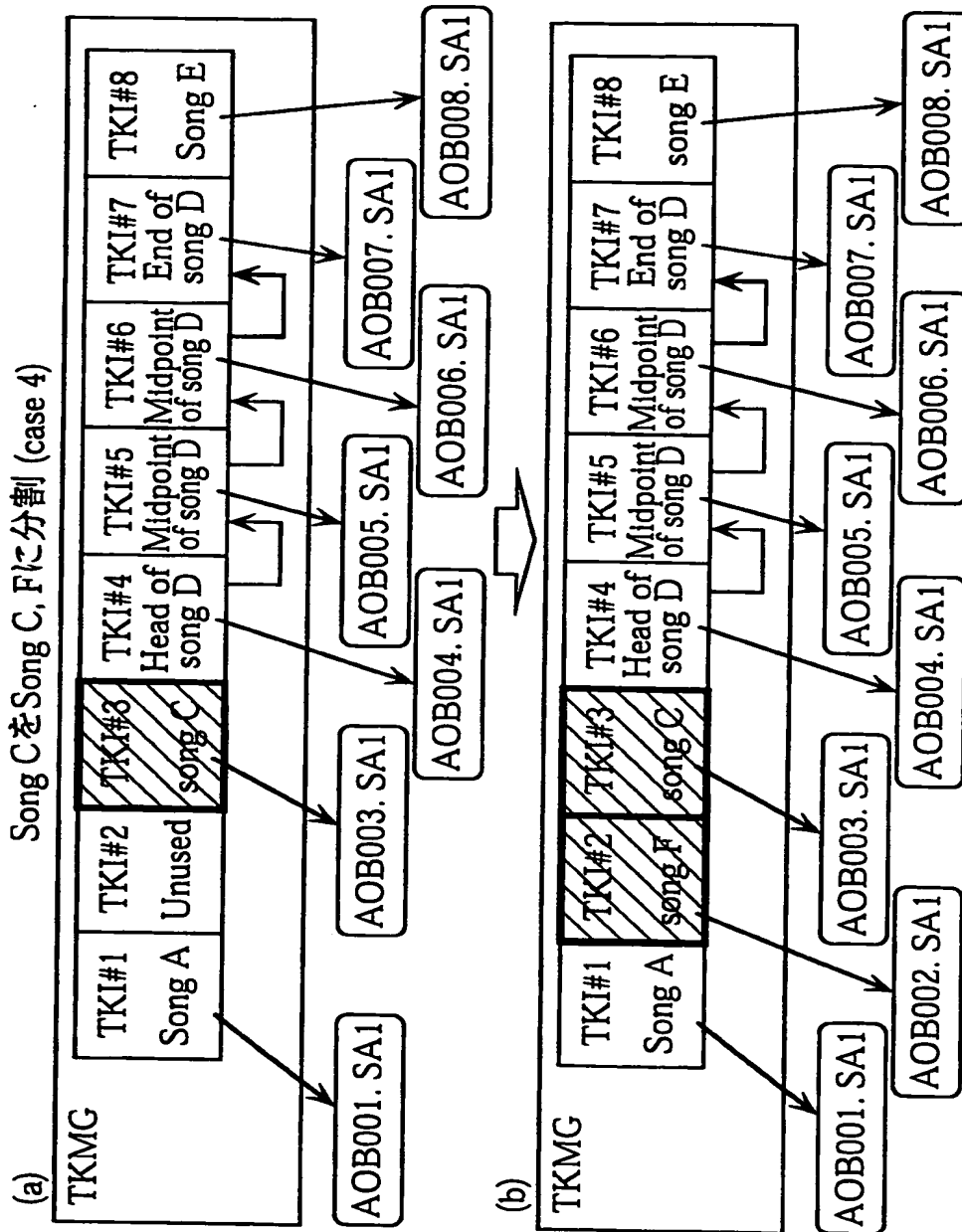
【図 2 8】



【図 2 9】



【図 3 0】

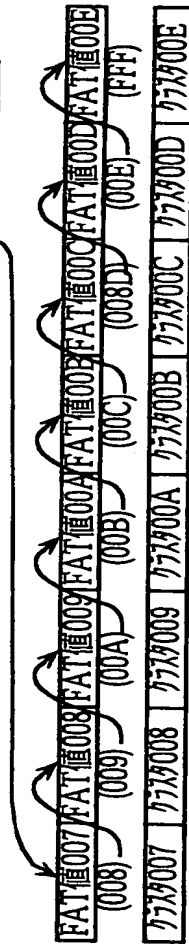


【図 3 1】

SD-Audio ディレクトリエントリ

(a)

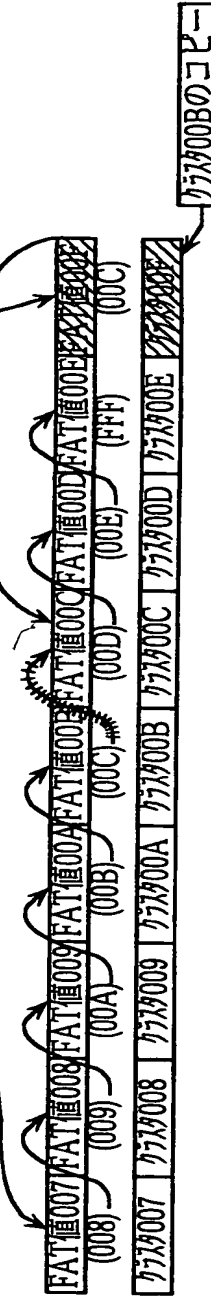
ファイル名	ファイル拡張子	ファイル最初のクラスタ番号
AOB003	SA 1	007



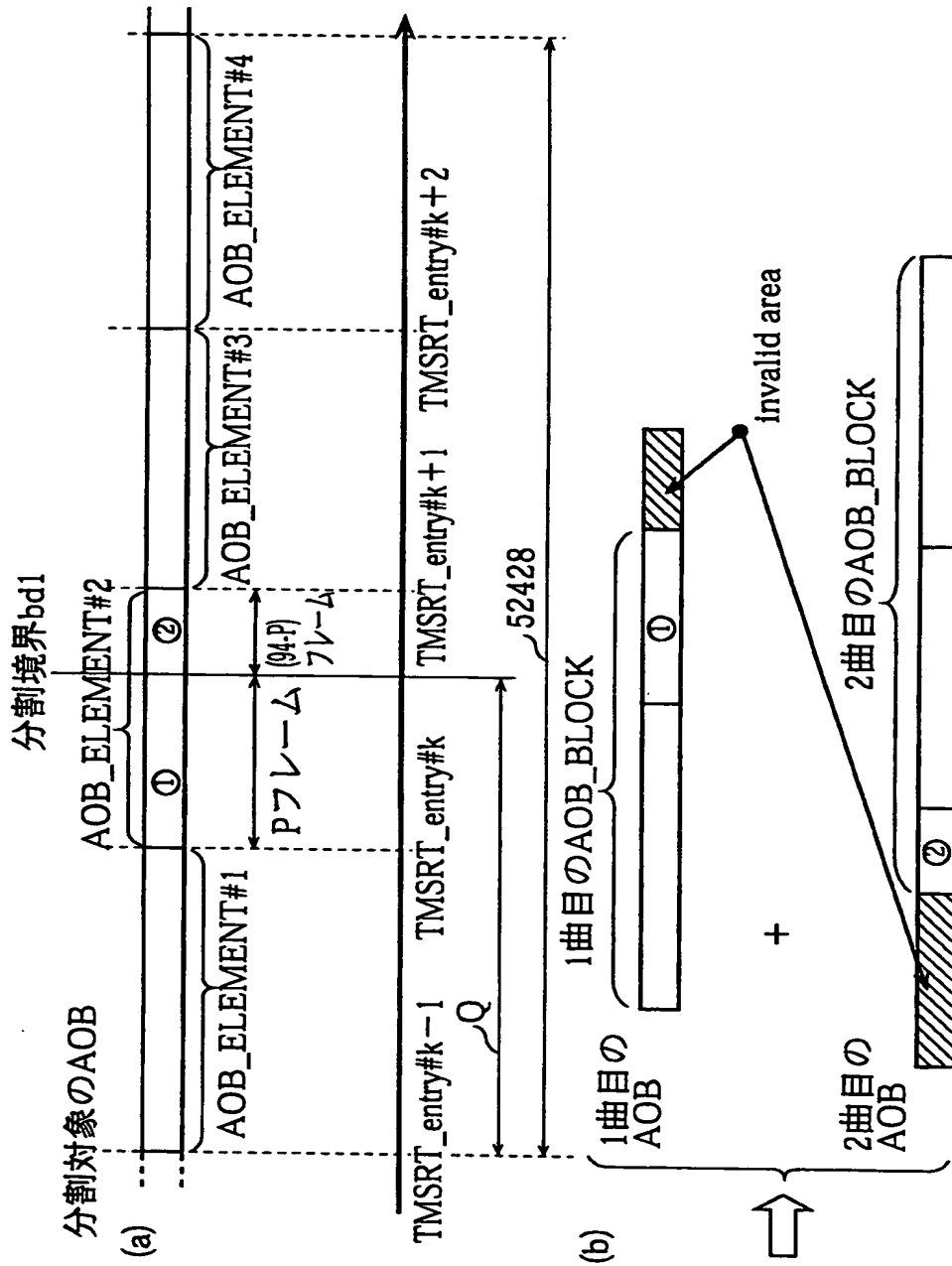
SD-Audio ディレクトリエントリ

(b)

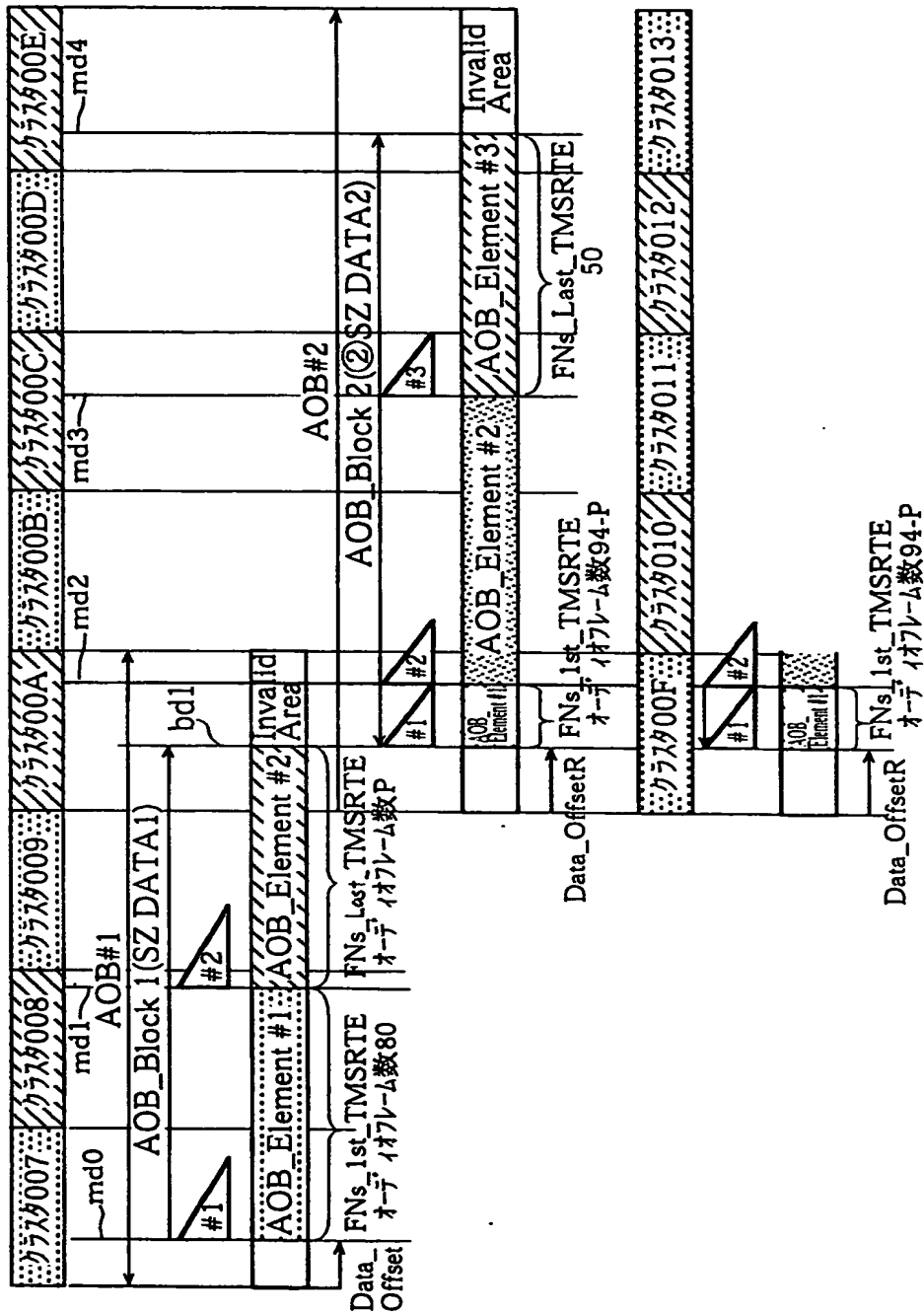
ファイル名	ファイル拡張子	ファイル最初のクラスタ番号
AOB002	SA 1	00F
ファイル名	ファイル拡張子	ファイル最初のクラスタ番号
AOB003	SA 1	007



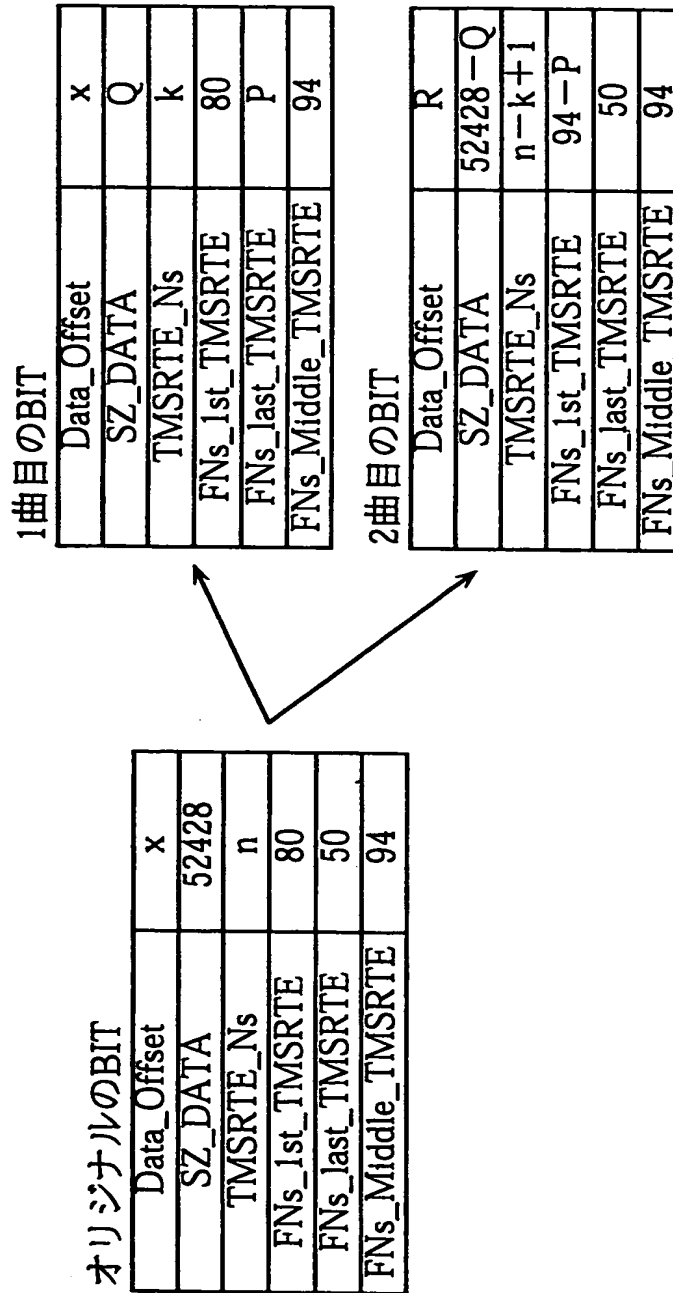
【図 3 2】



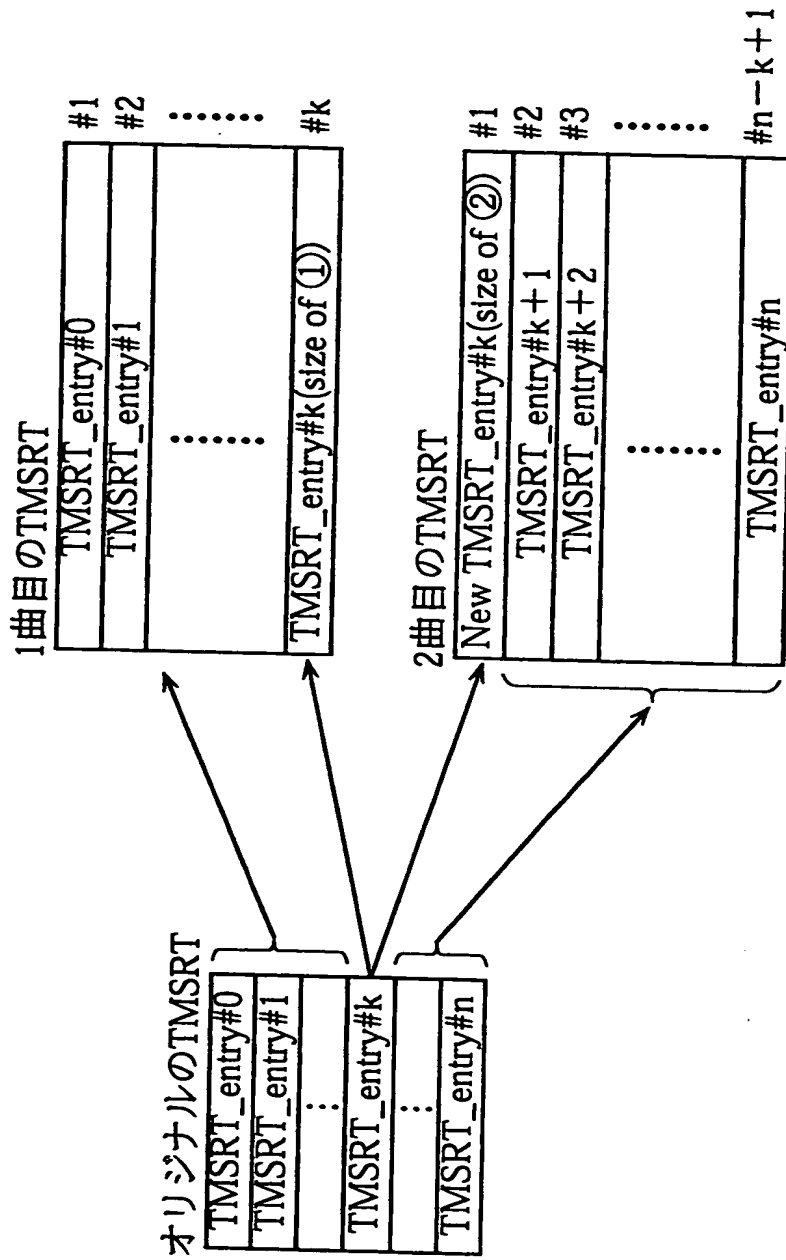
【図 3 3】



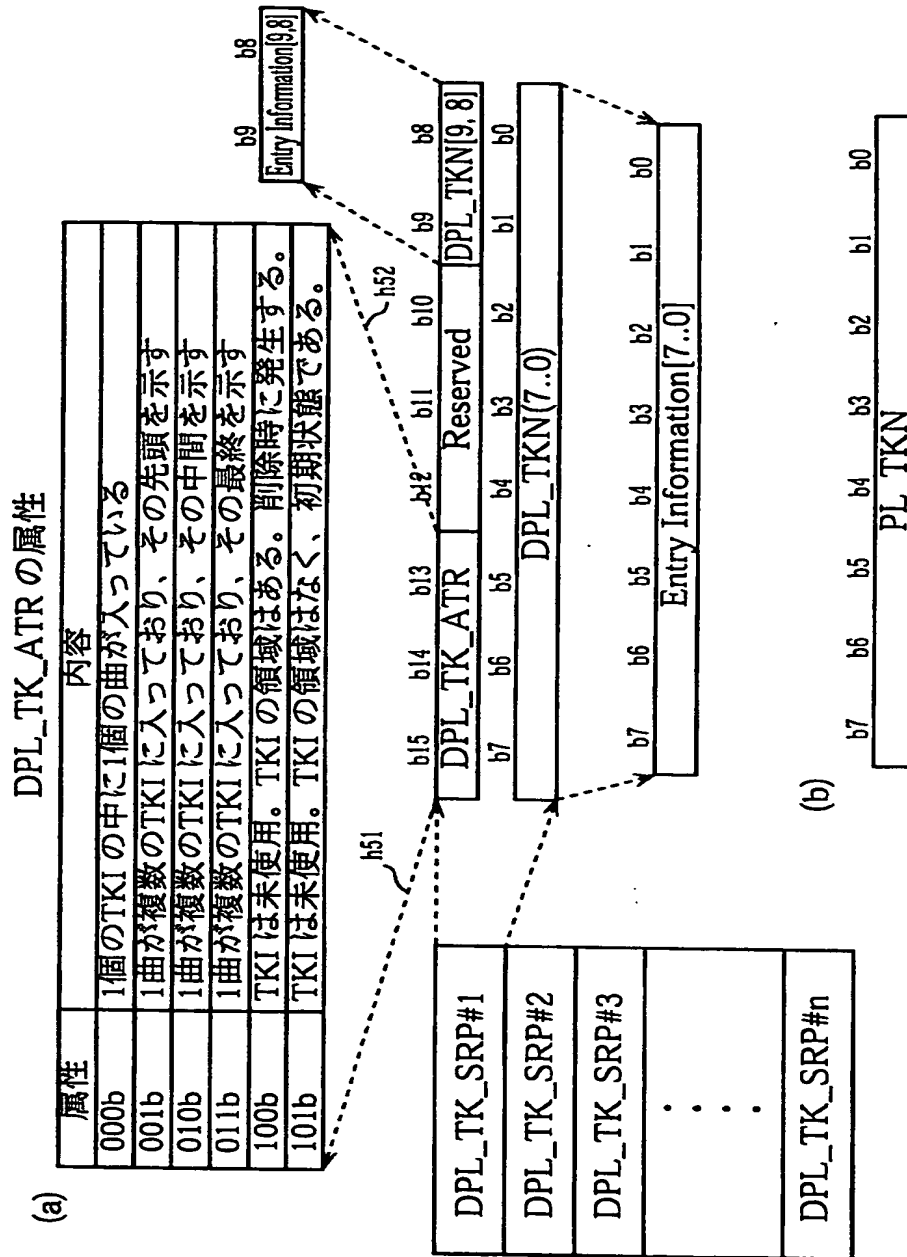
【図 3 4】



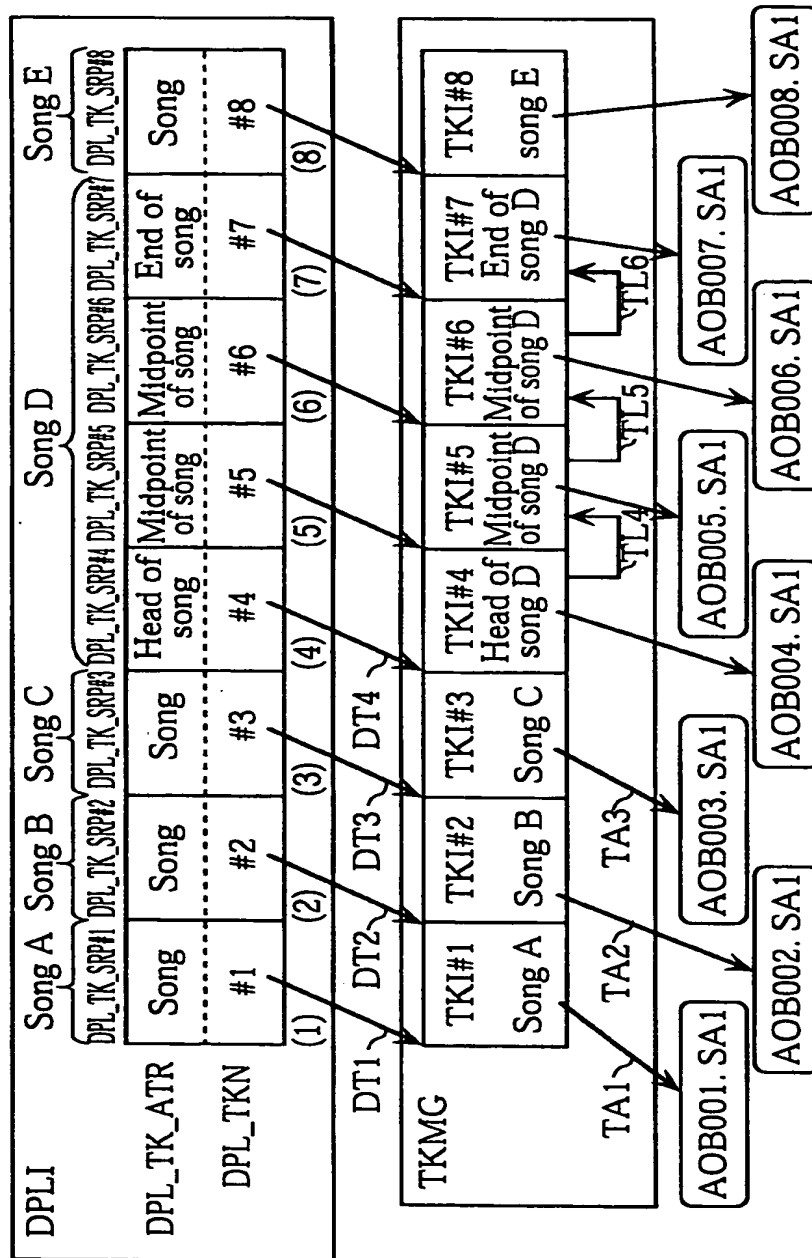
【図 3 5】



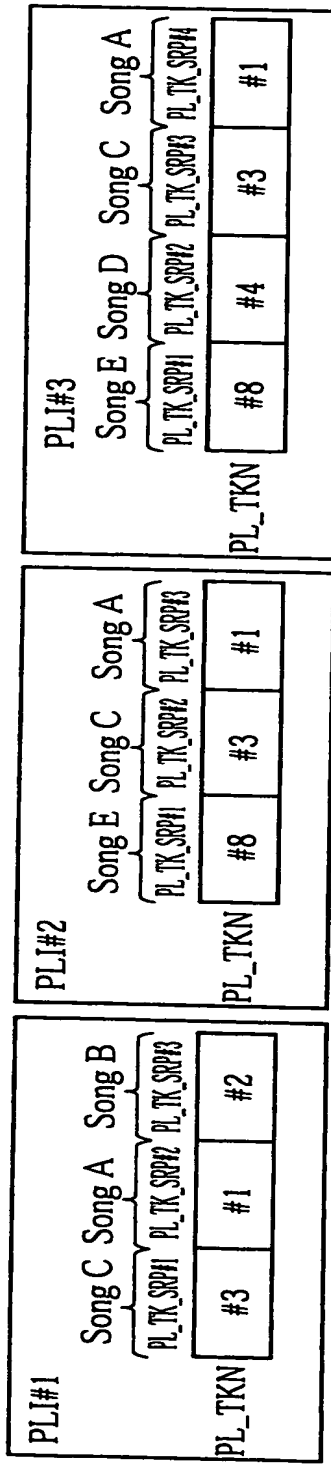
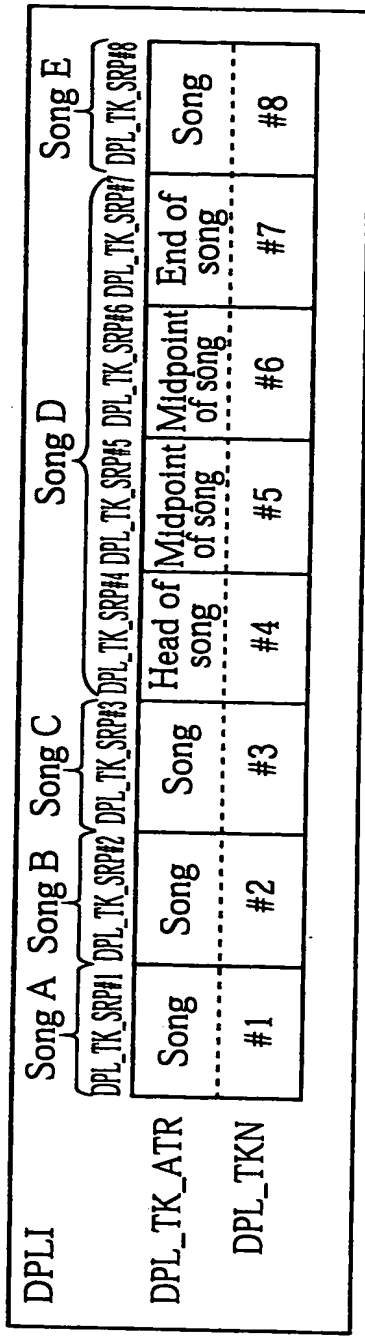
【図 3 6】



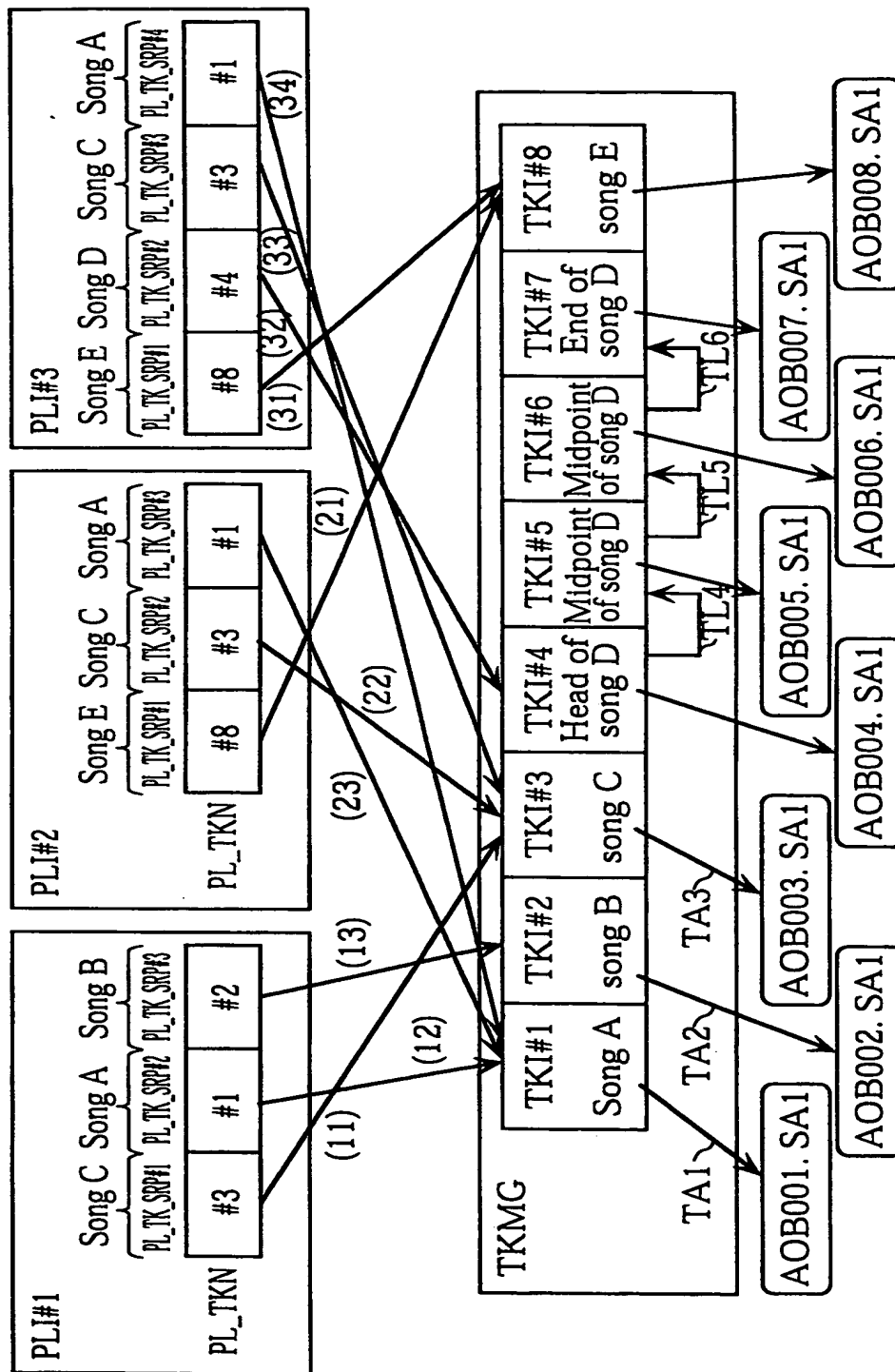
【圖 3 7】



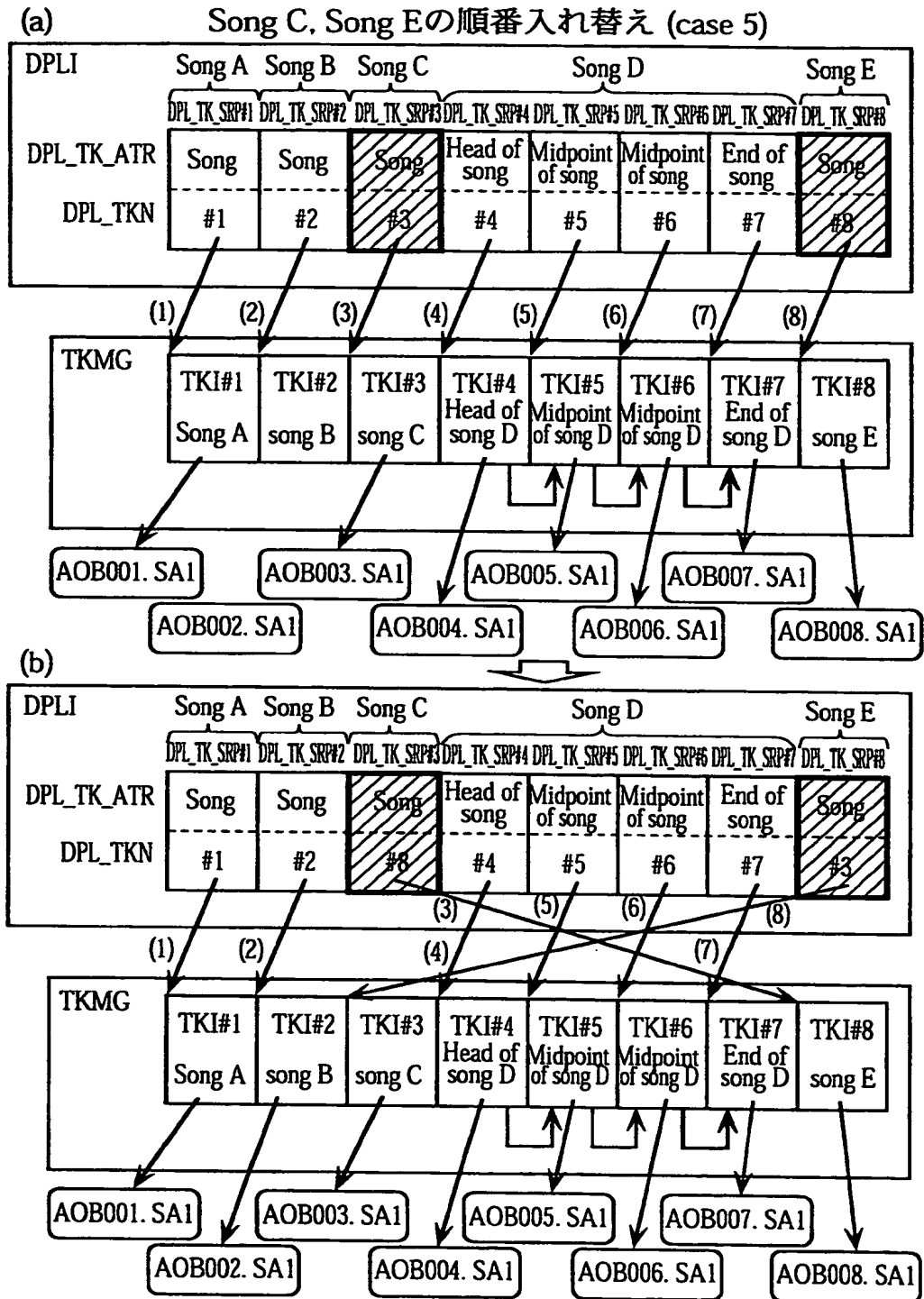
【図 3 8】



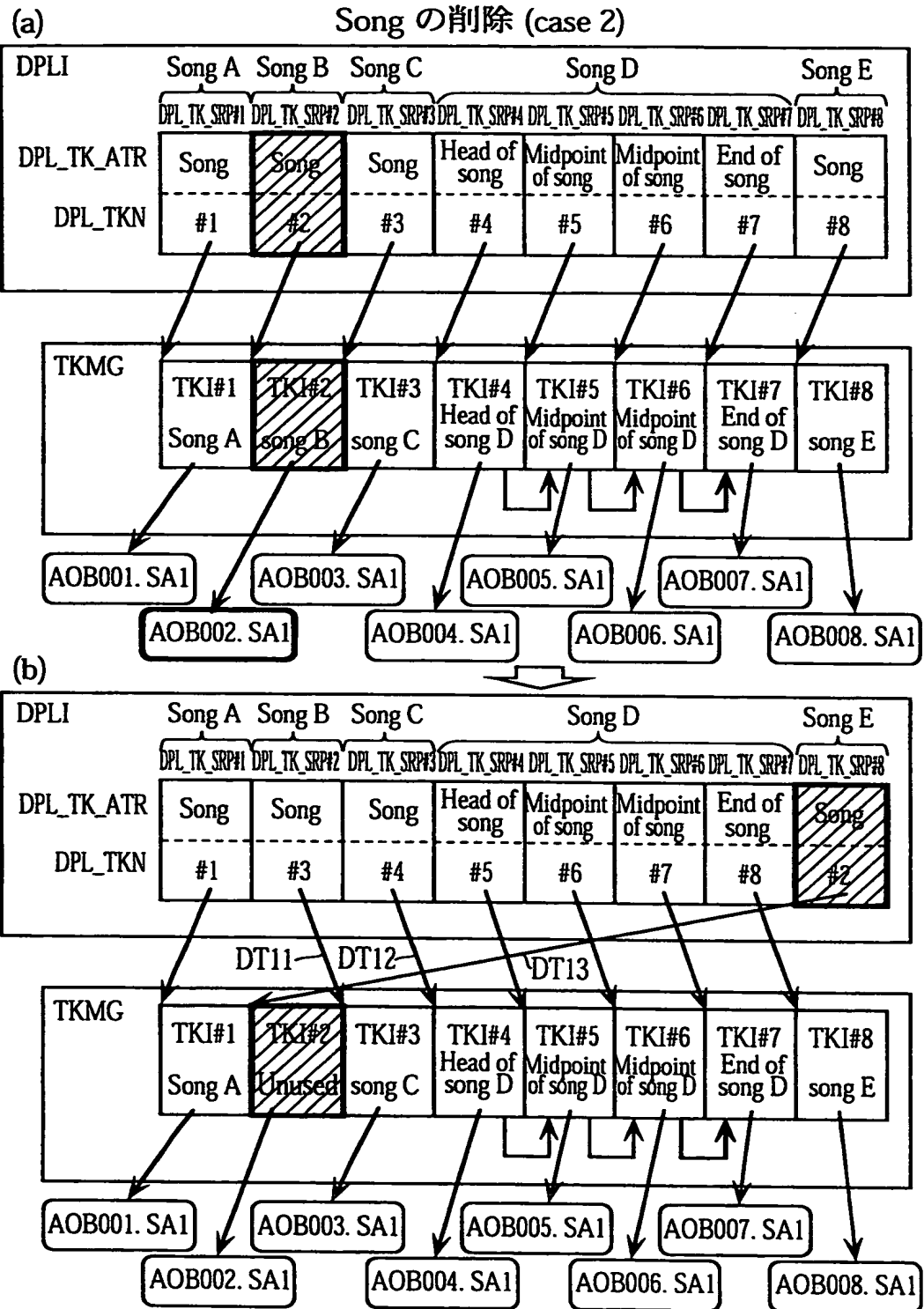
【図 3 9】



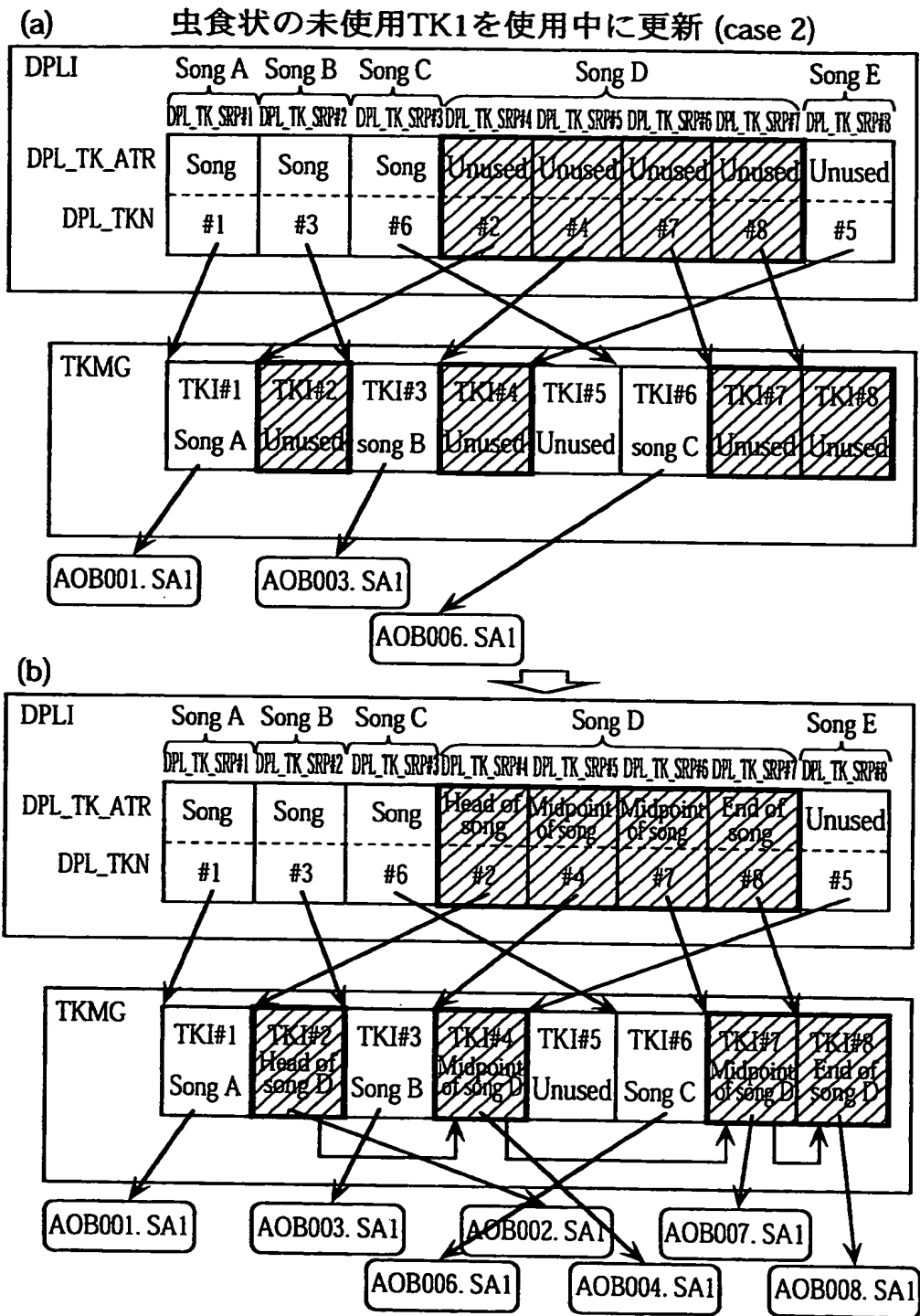
【図 4 0】



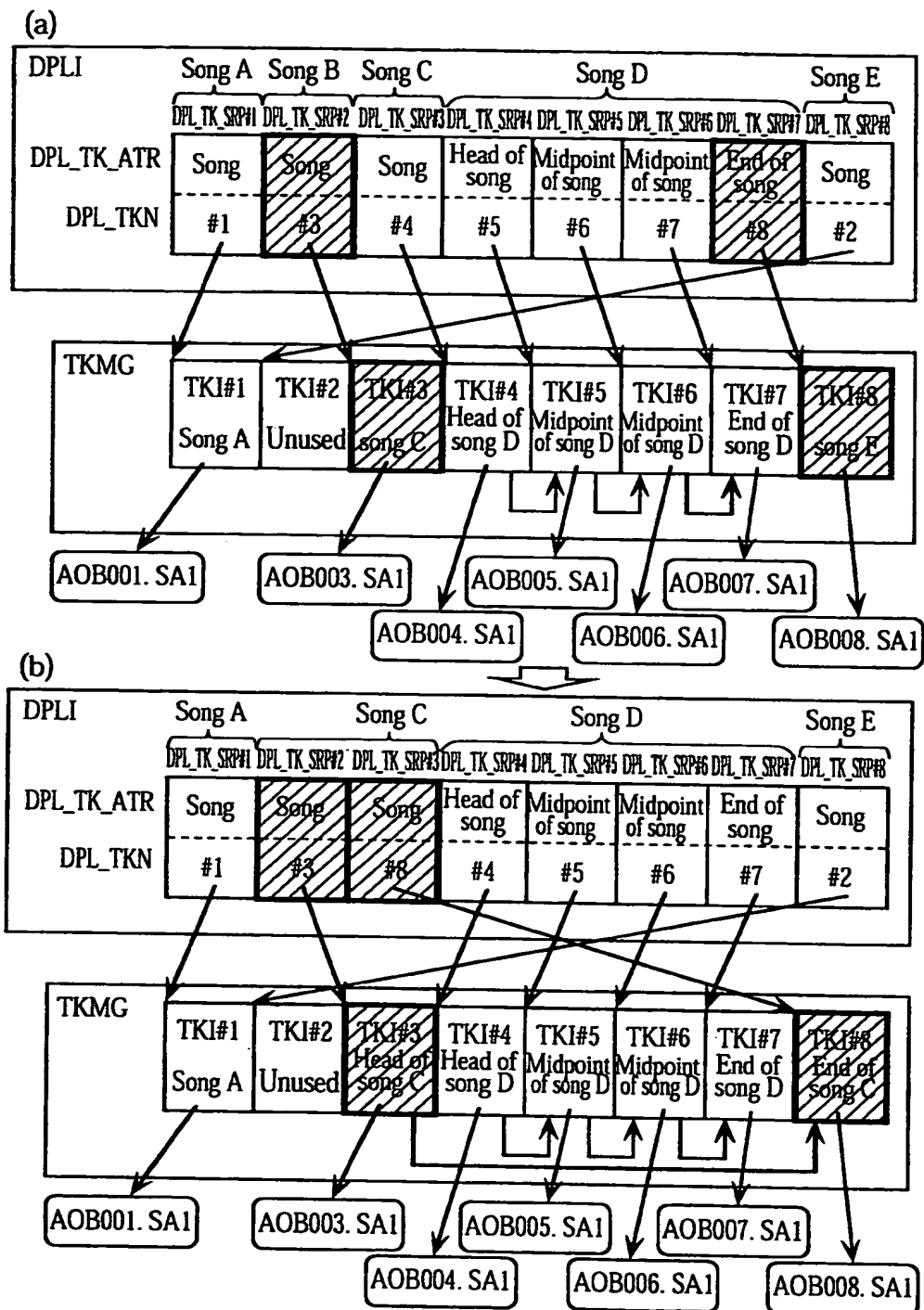
【図 4 1】



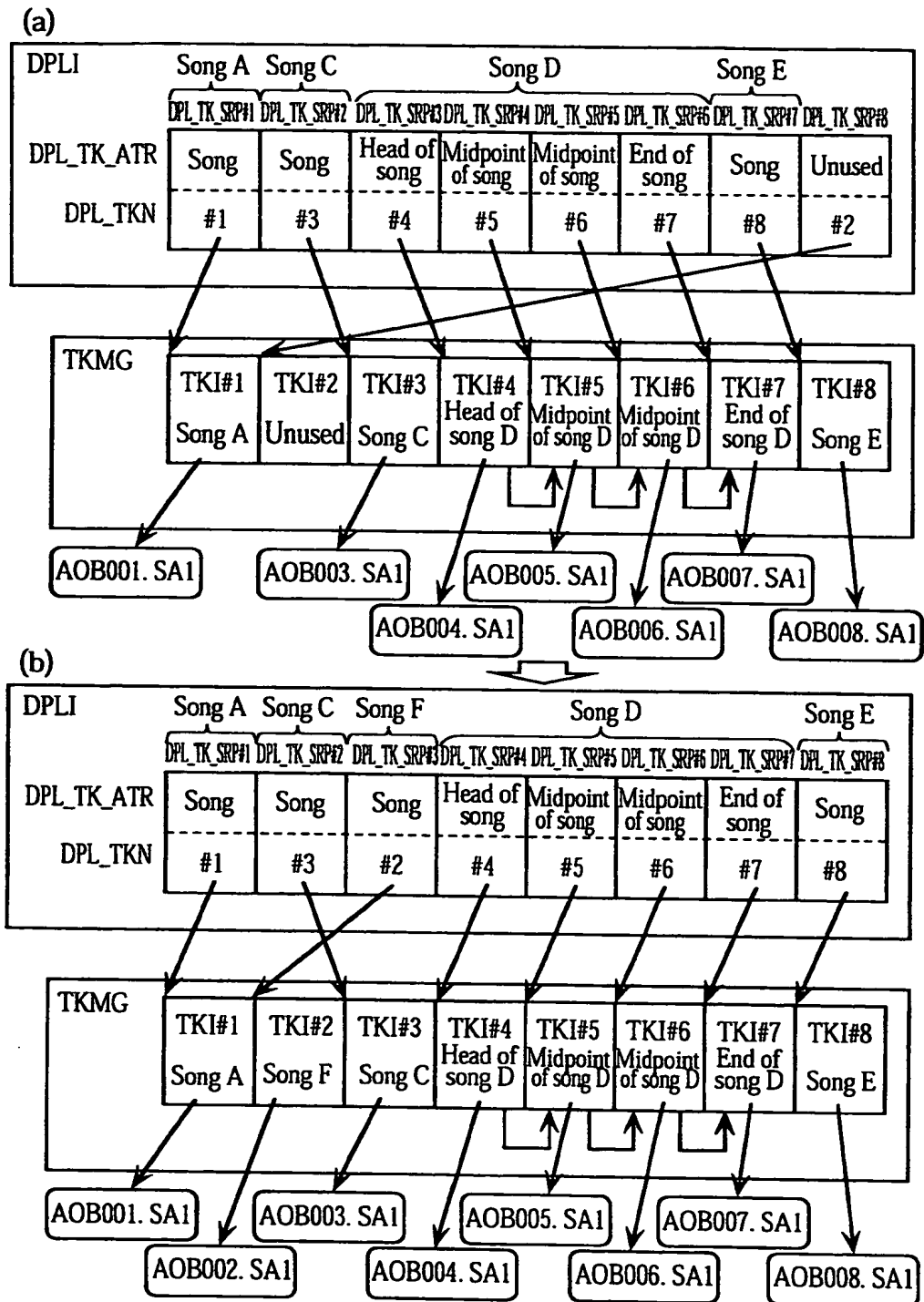
【図 4 2】



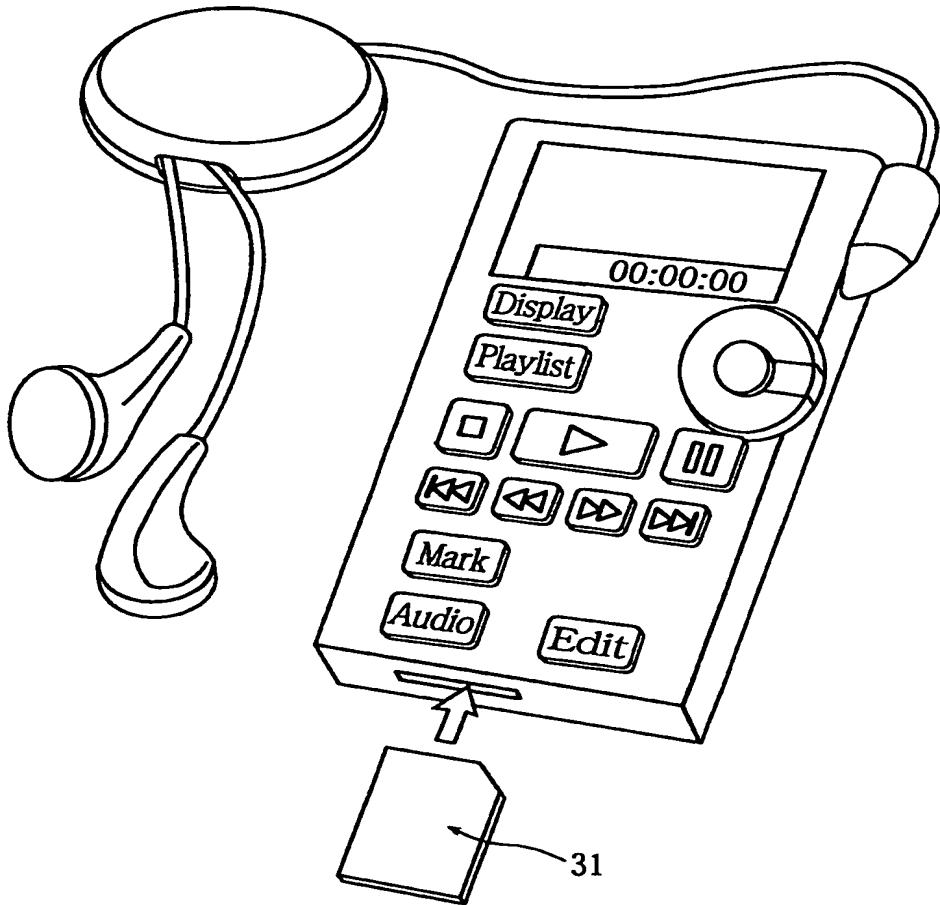
【図 4 3】



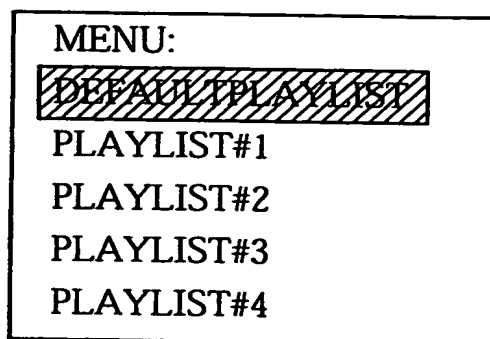
【図 4 4】



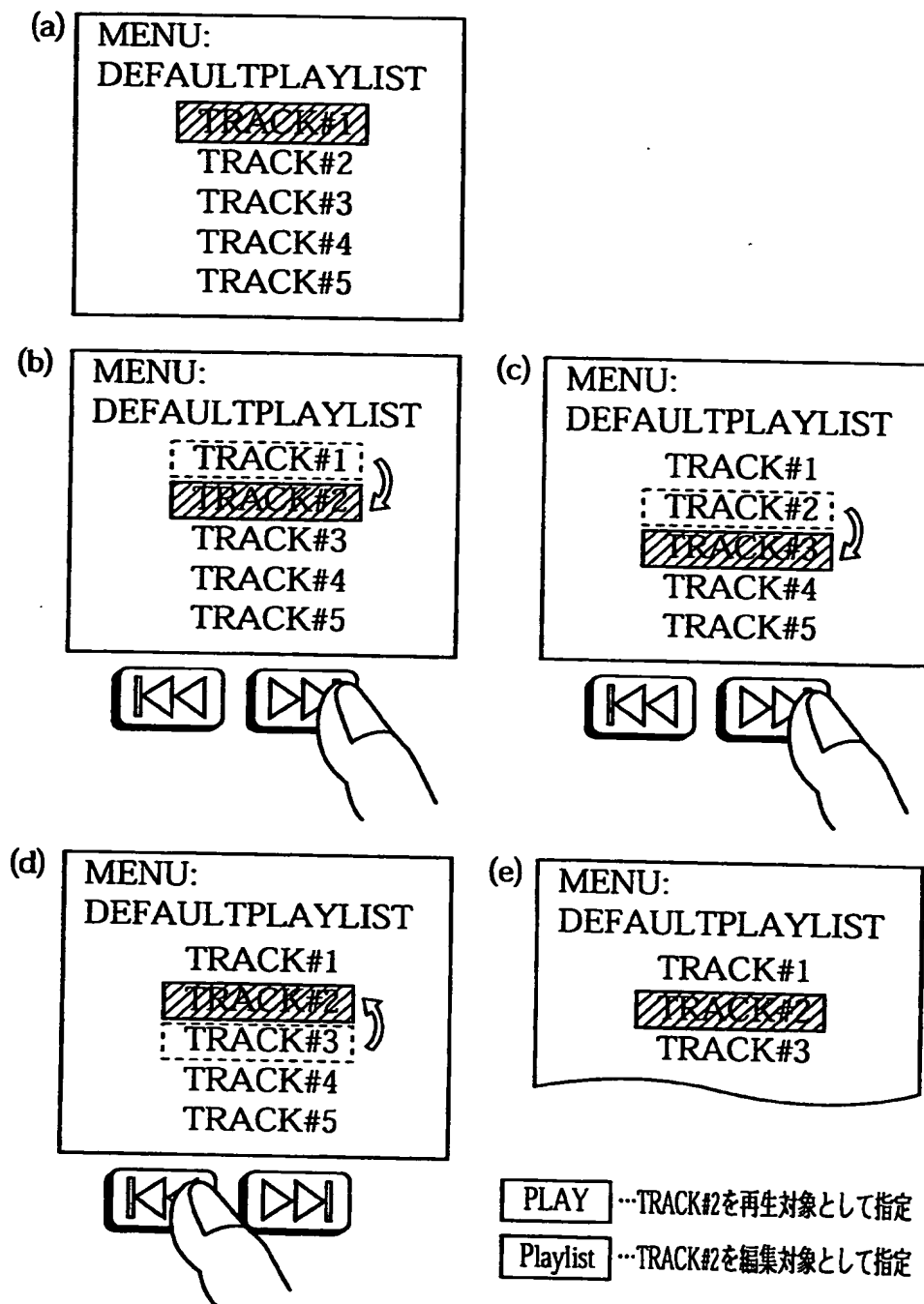
【図 4 5】



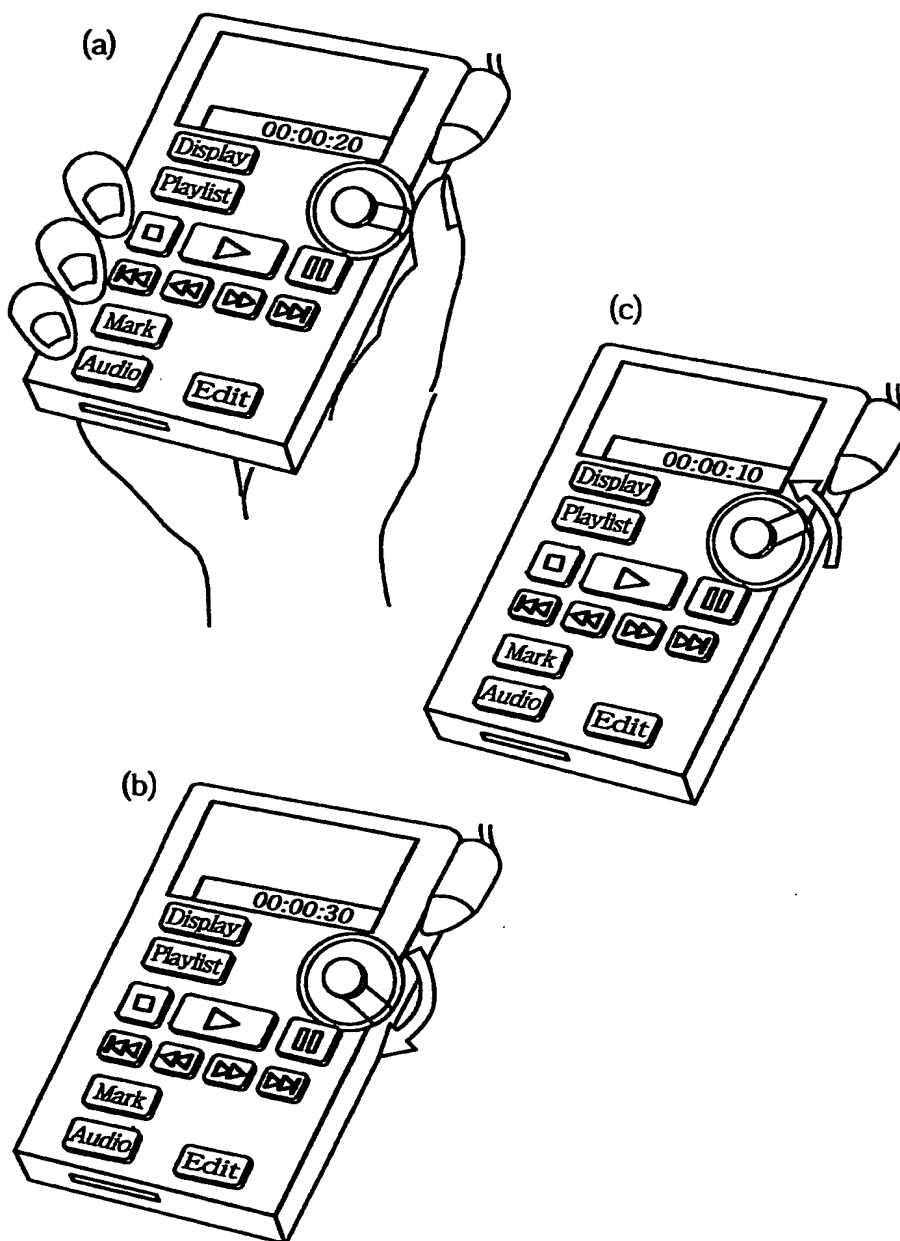
【図 4 6】



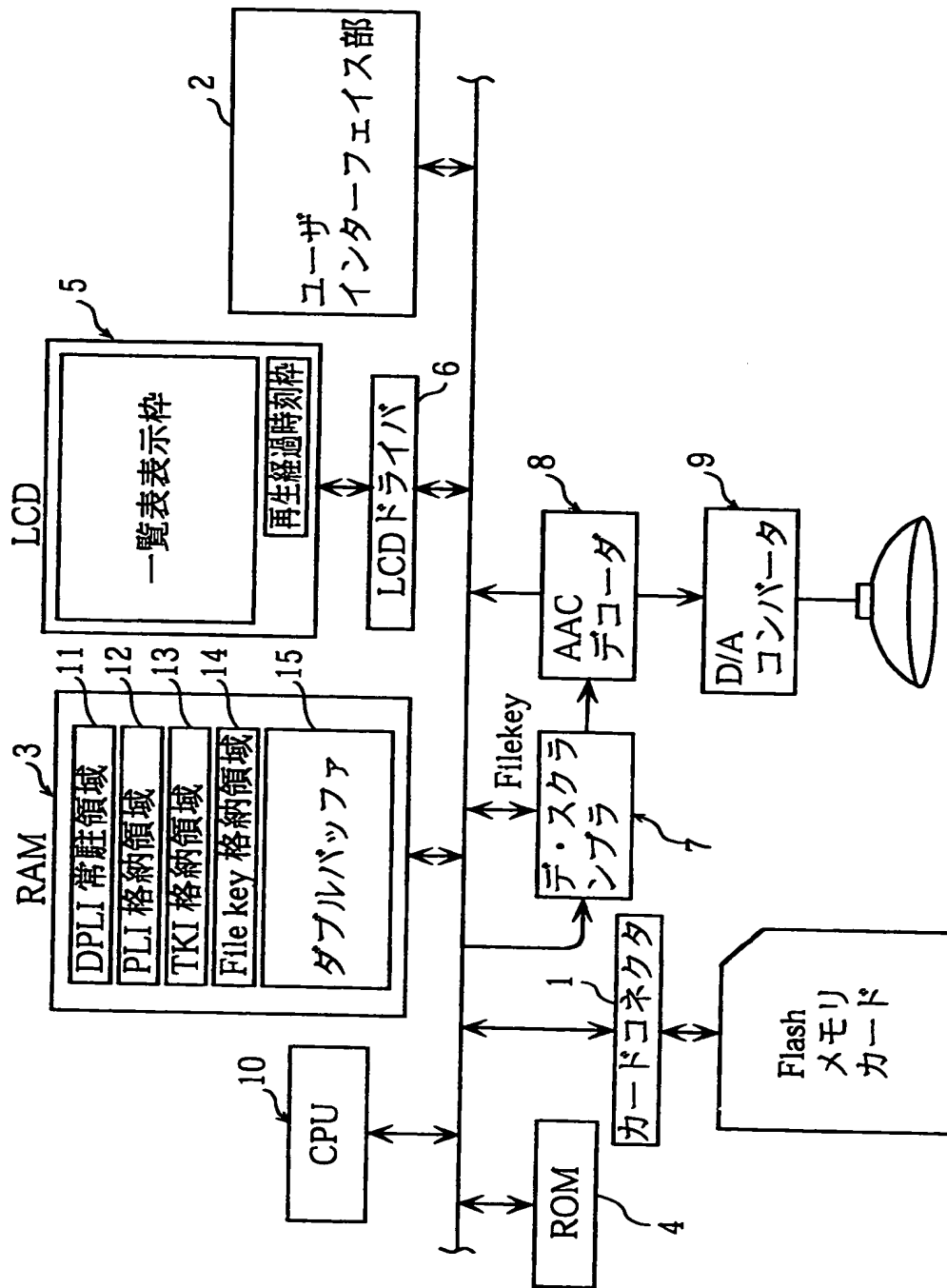
【図 4 7】



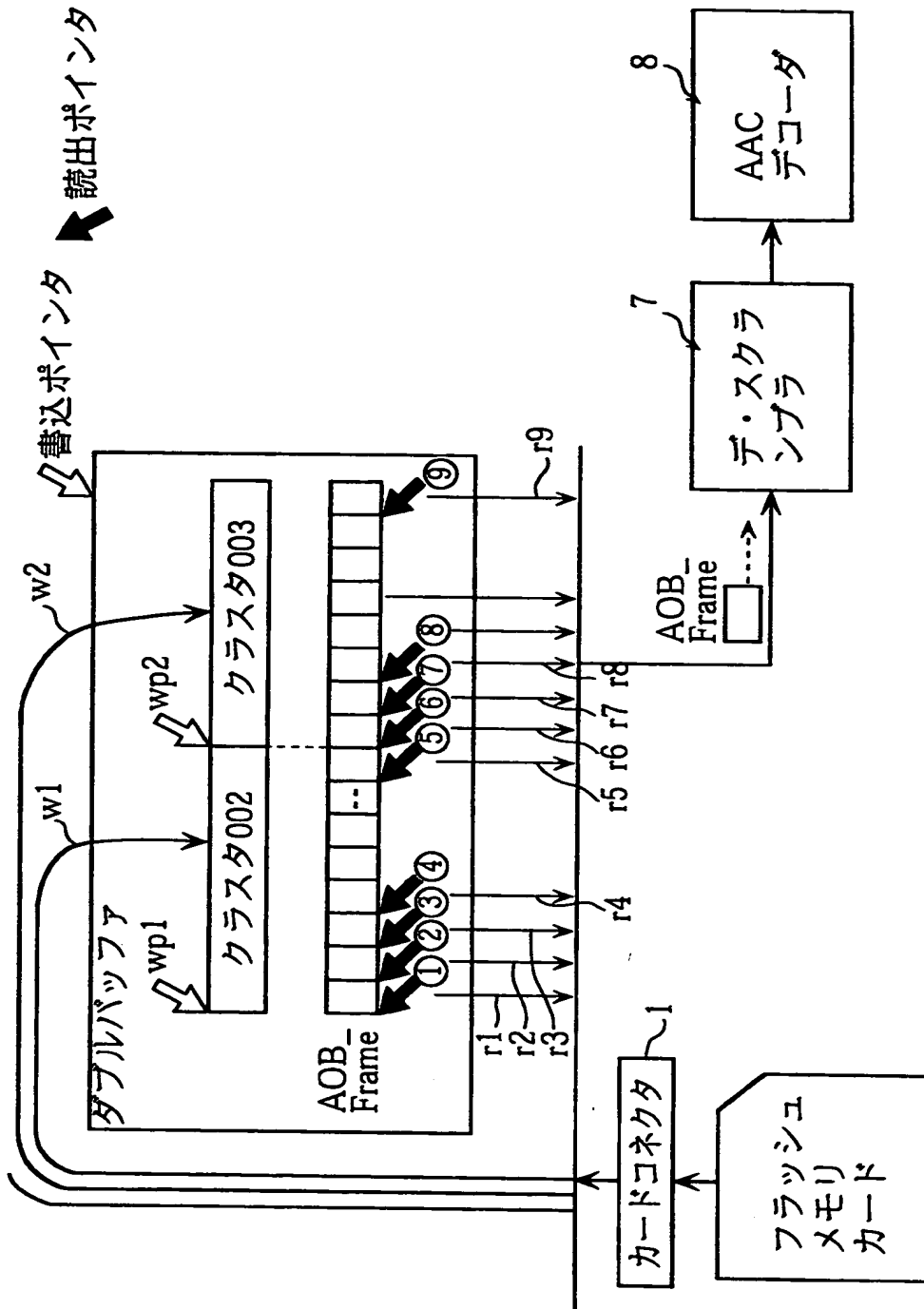
【図 4 8】



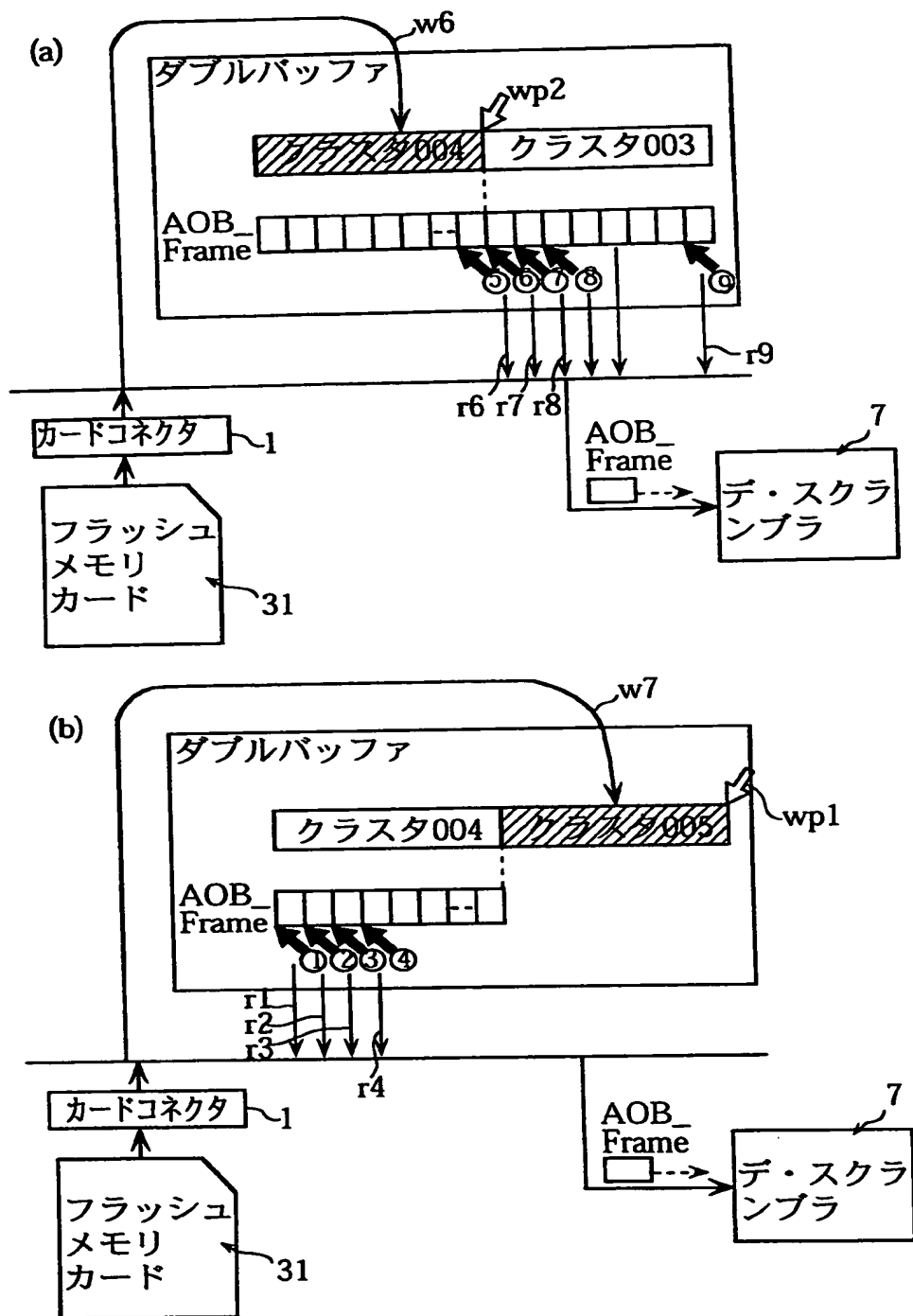
【図 49】



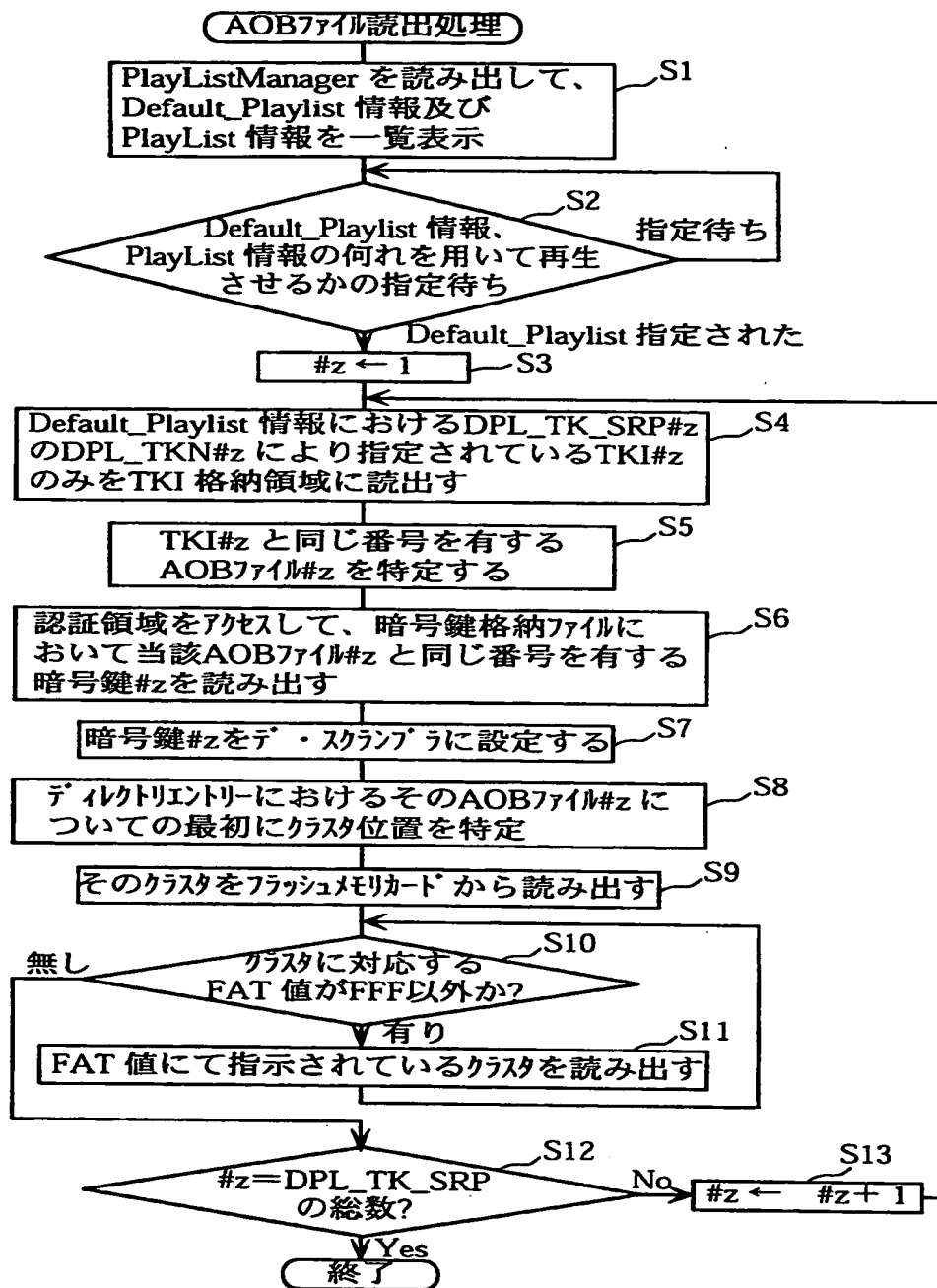
【図 5 0】



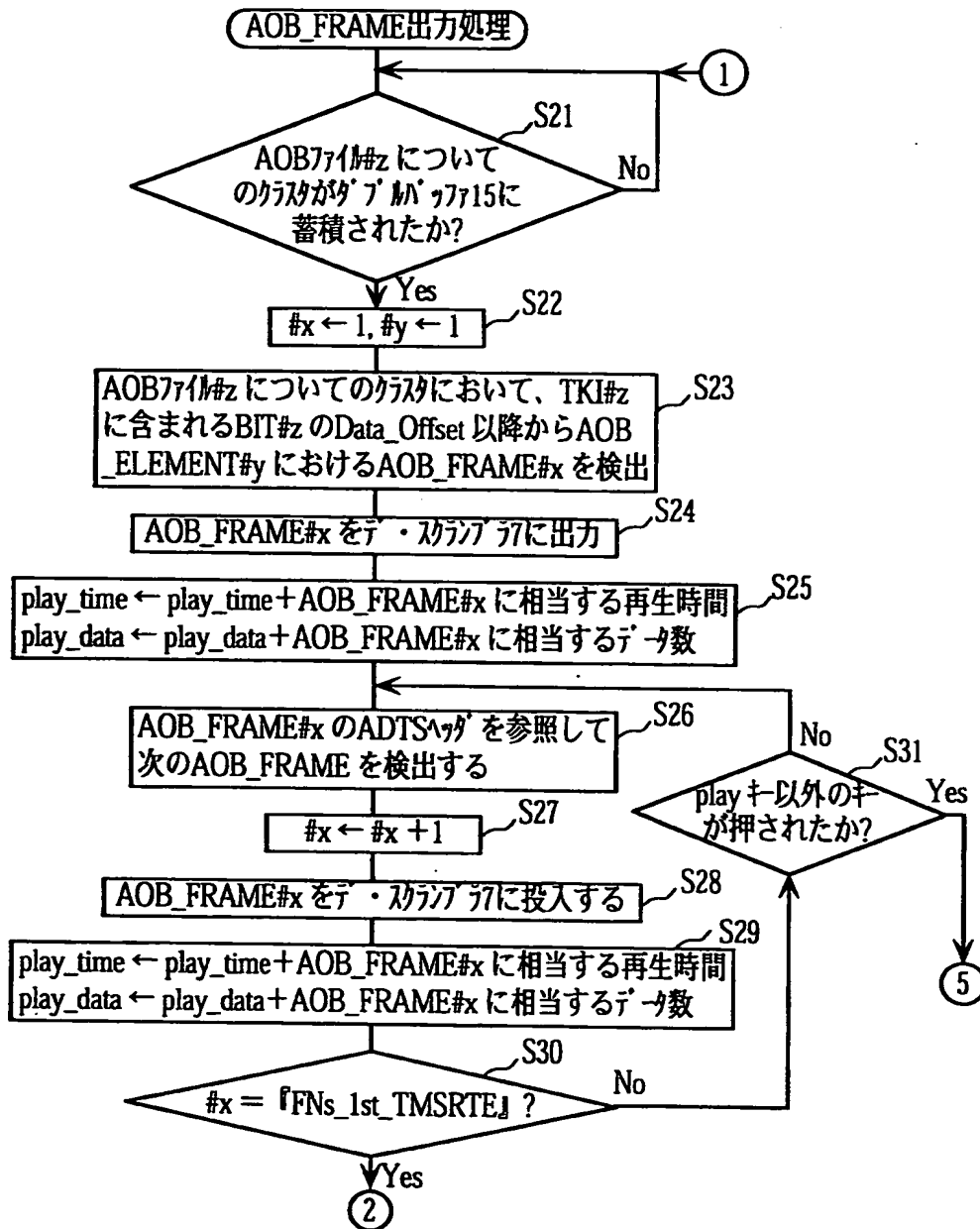
【図 5 1】



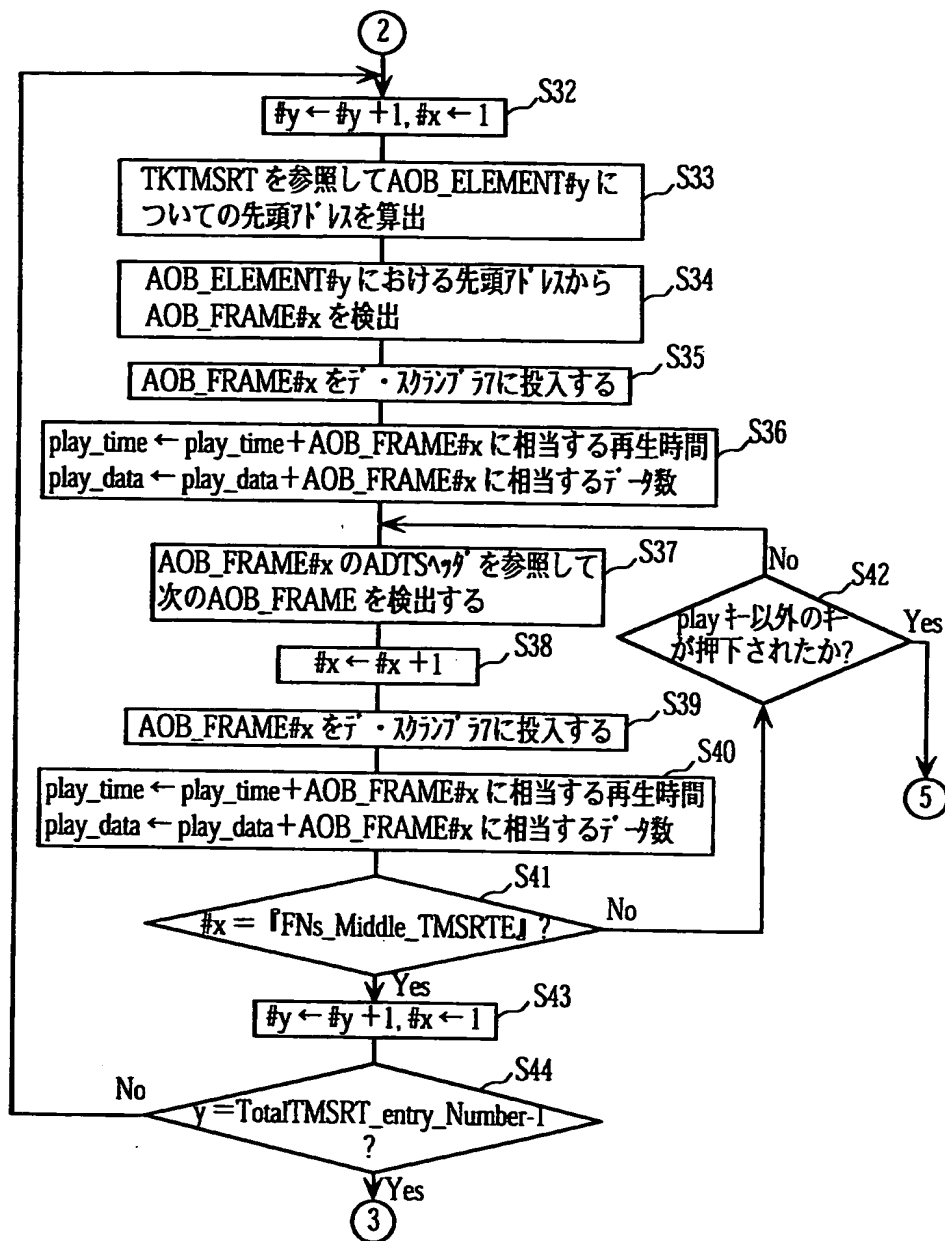
【図 5 2】



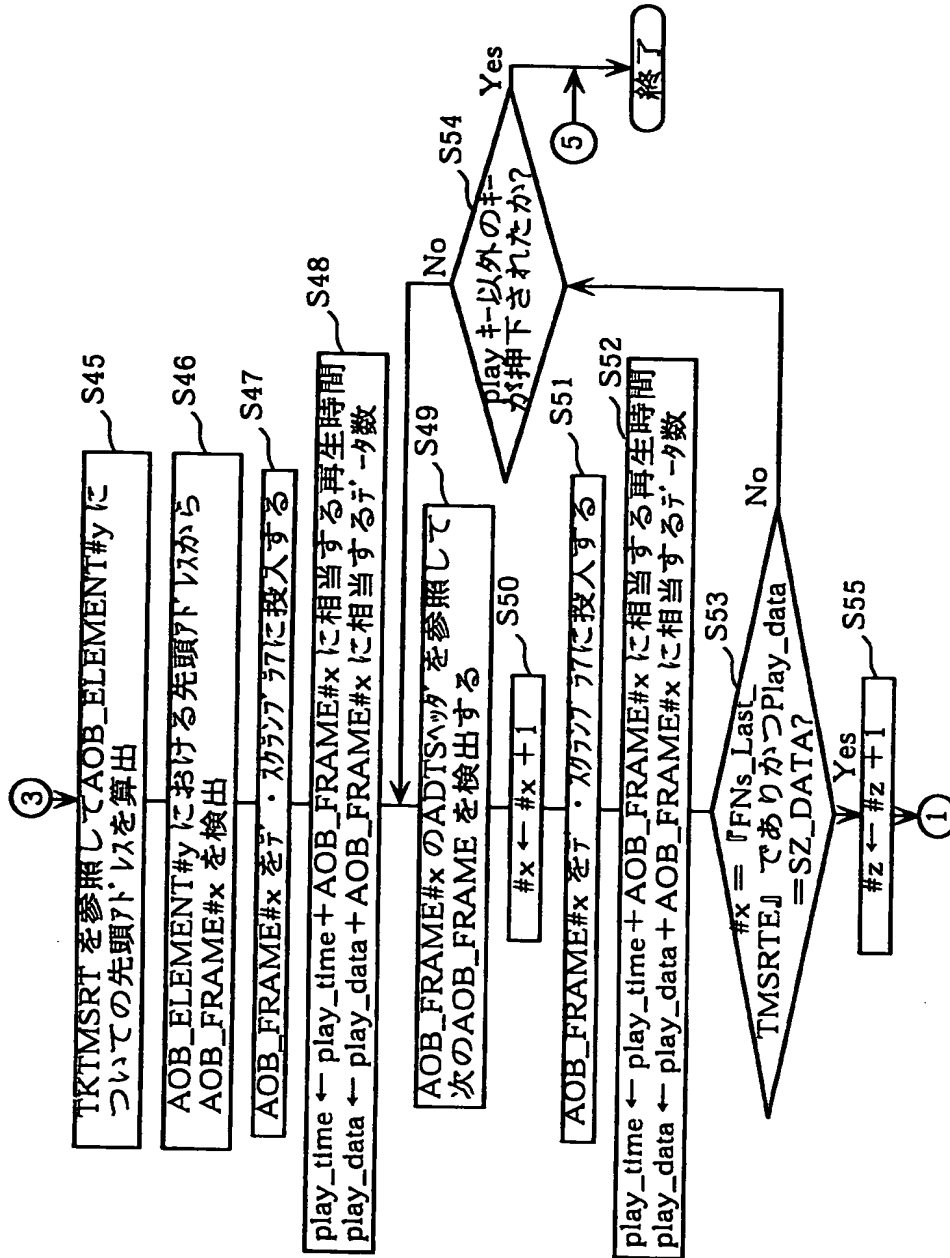
【図 5 3】



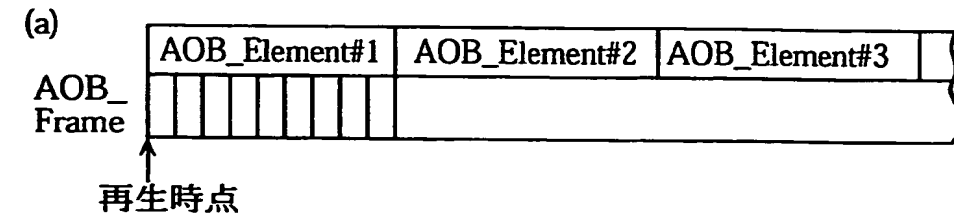
【図 5 4】



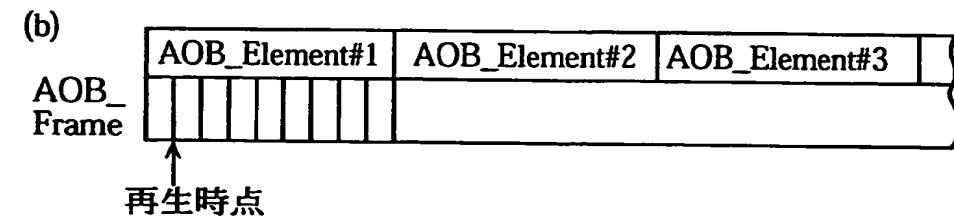
【図 5 5】



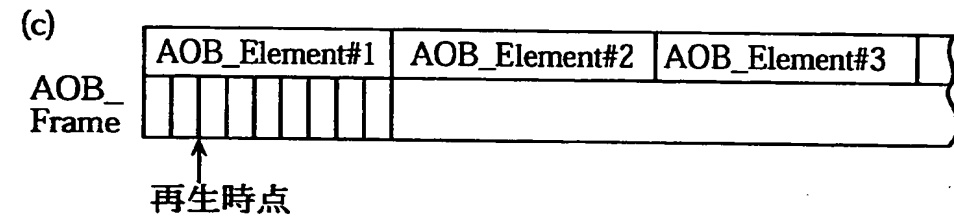
【圖 5 6】



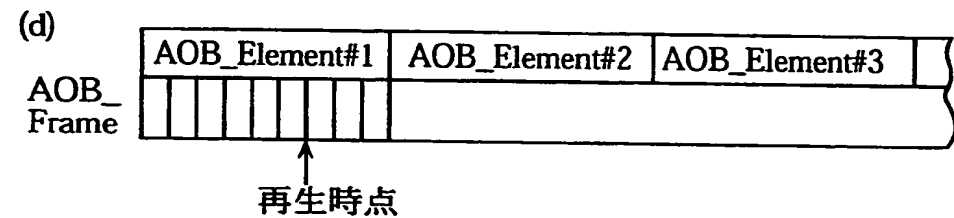
再生經過時刻=00:00:00.000



再生經過時刻=00:00:00.020

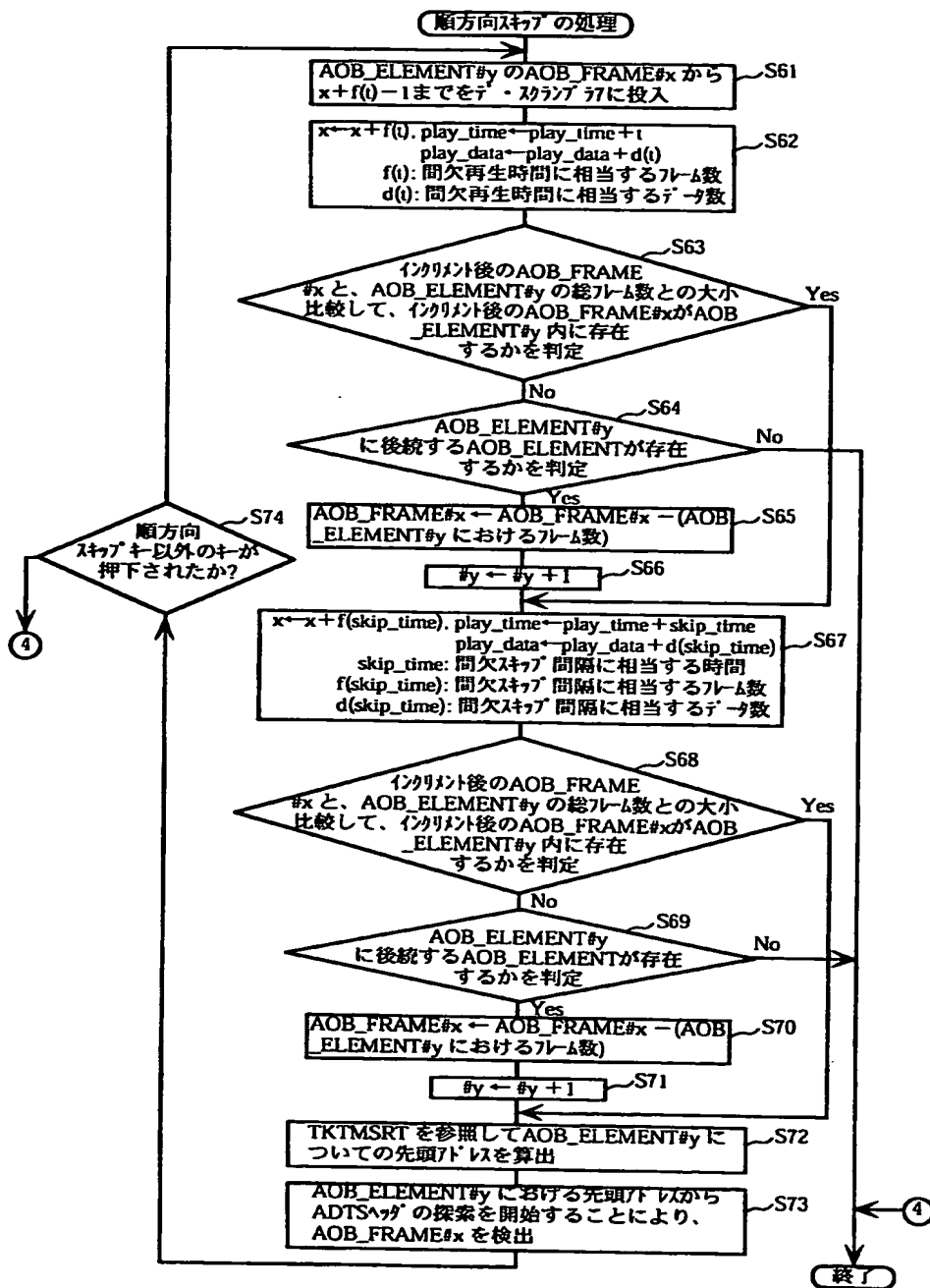


再生經過時刻=00:00:00.040

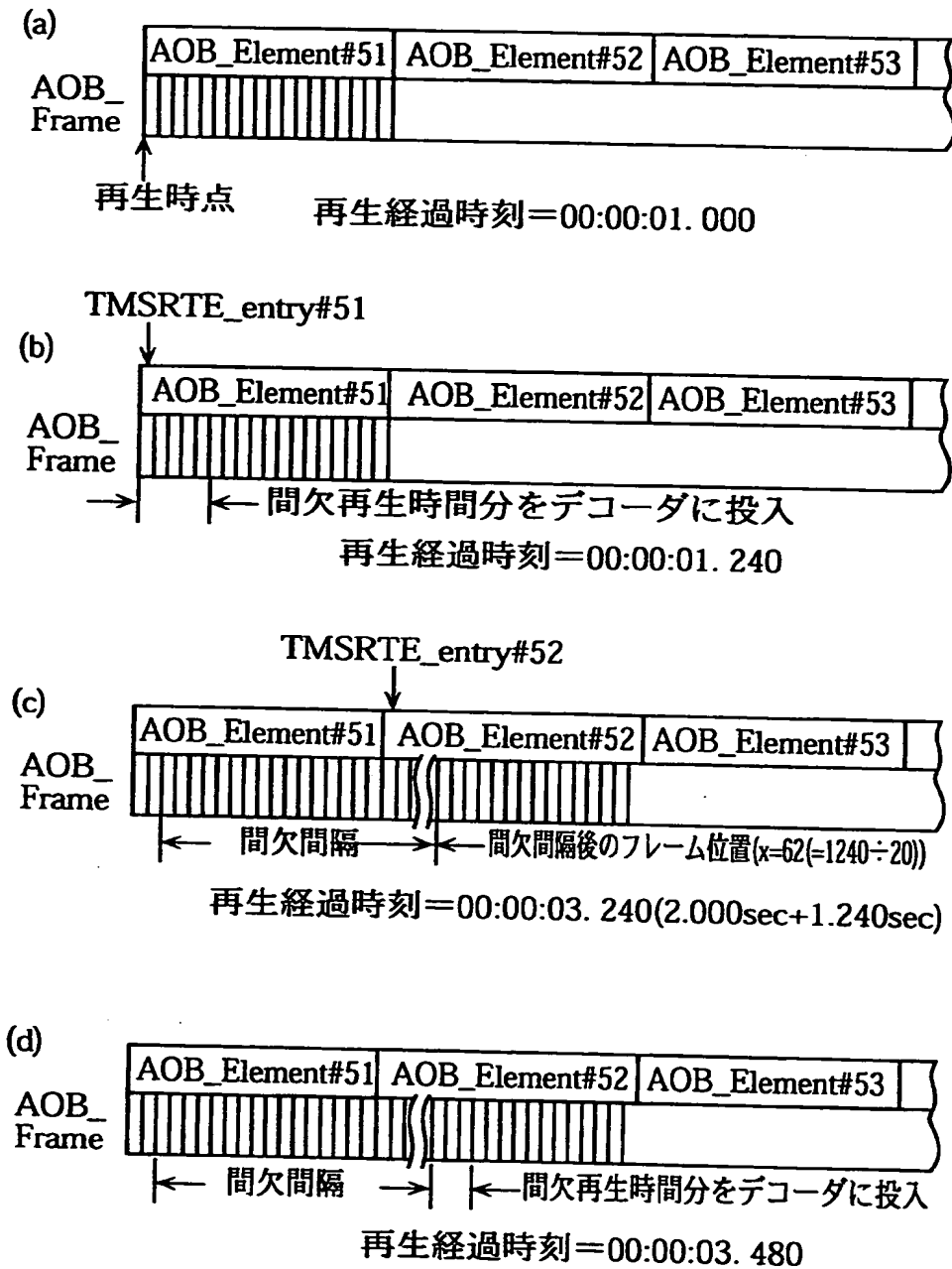


再生經過時刻=00:00:00.120

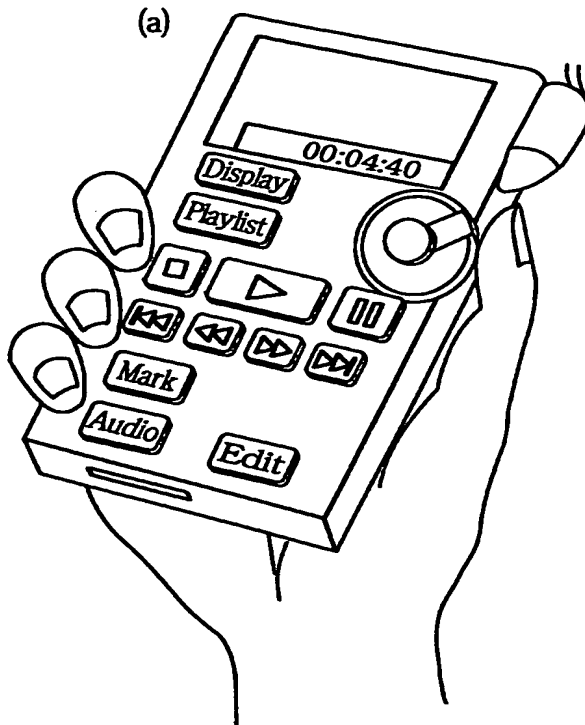
【図 5 7】



【図 5 8】

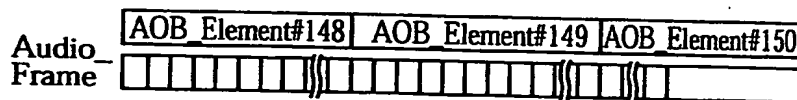


【図 5 9】



(b)

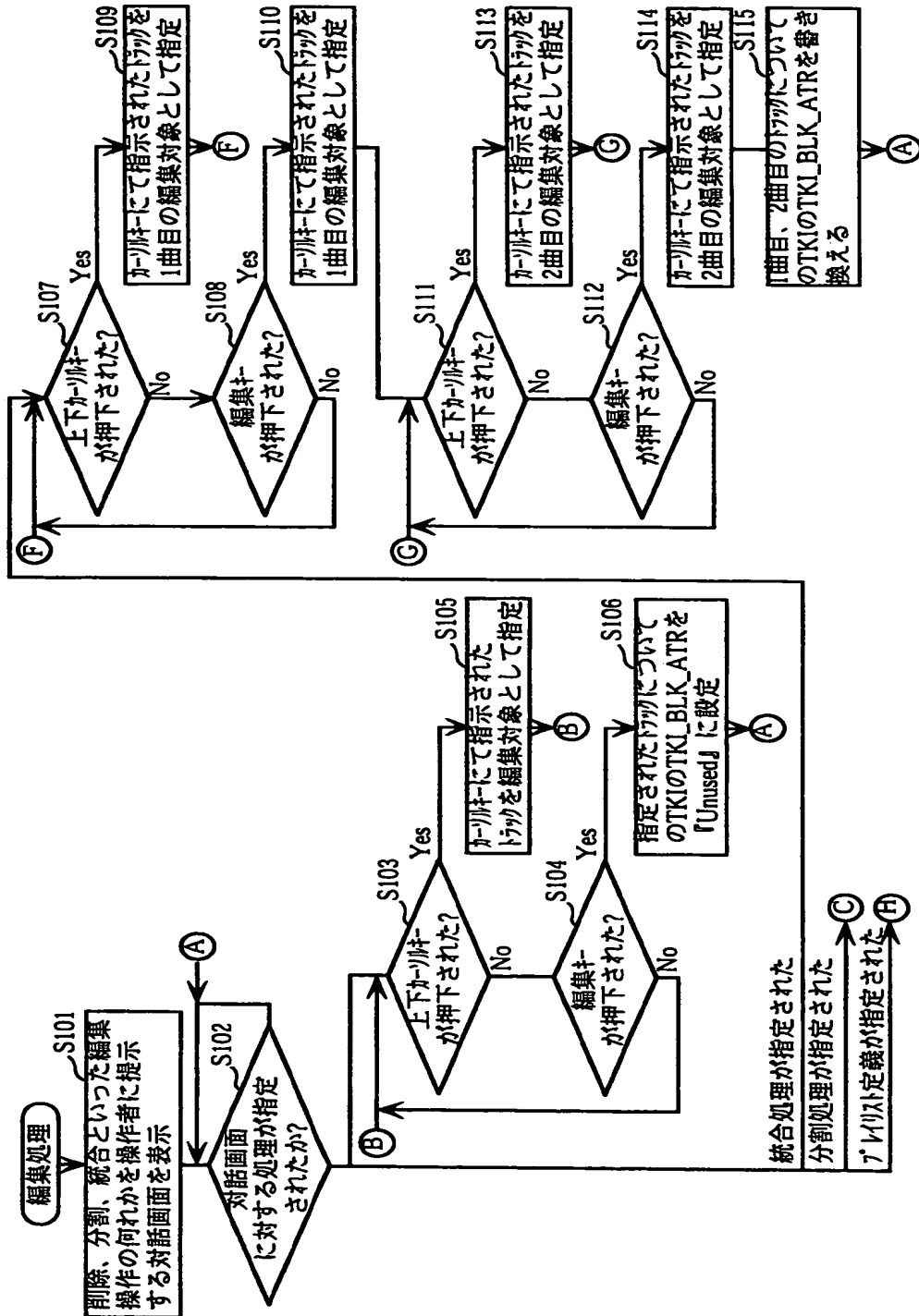
BIT	FNs_1st_TMSRTE	80
	FNs_Last_TMSRTE	50
	FNs_Middle_TMSRTE	94



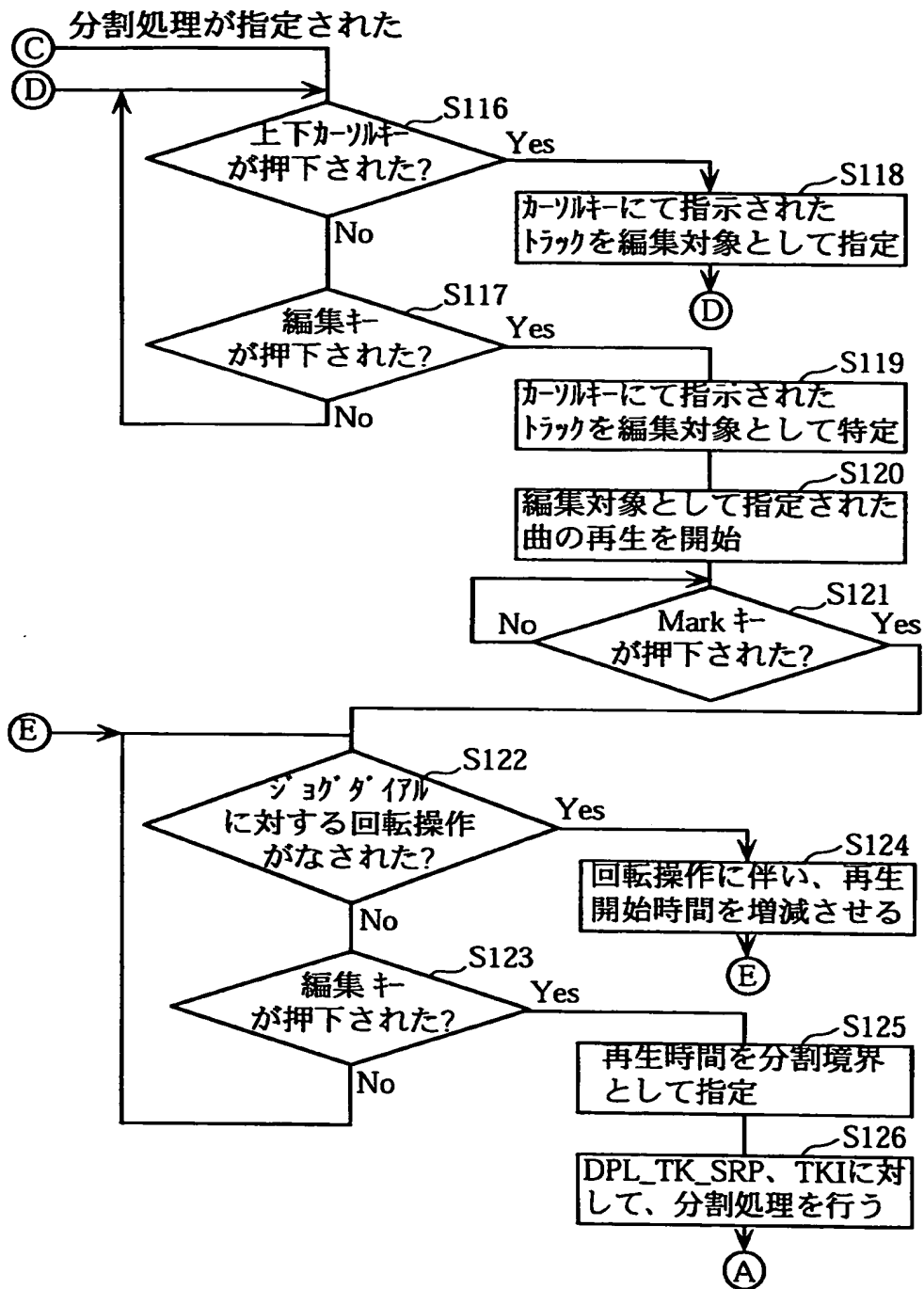
再生経過時刻=00:04:40.000(=280sec)
 $280\text{sec} = (80(=\text{FNs_1st_TMSRTE}) + 148 \times 94$
 $(=\text{FNs_middle_TMSRTE}) + 8) \times 20\text{msec}$

再生を開始すべき
 フレーム位置
 (AOB_Element#150
 の8フレーム目)

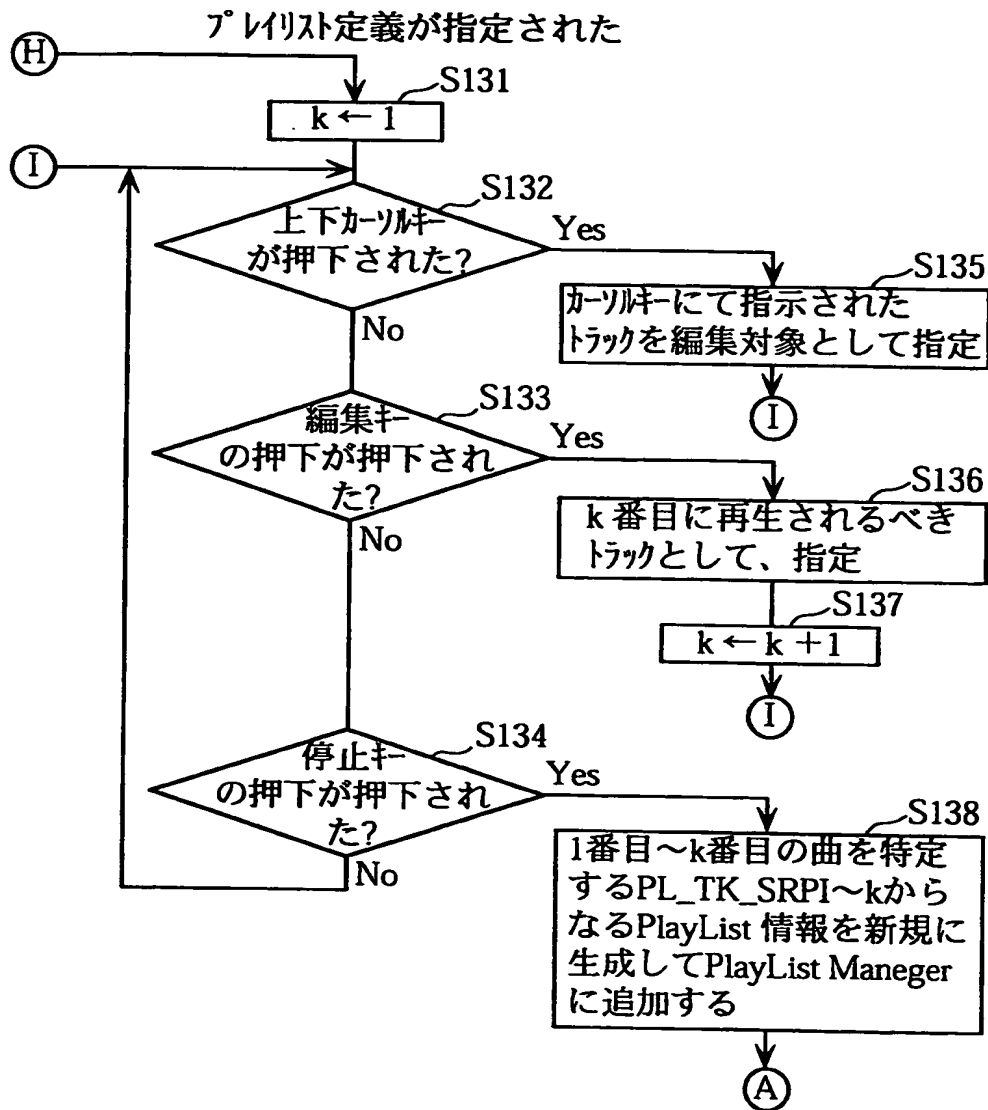
【図 6 0】



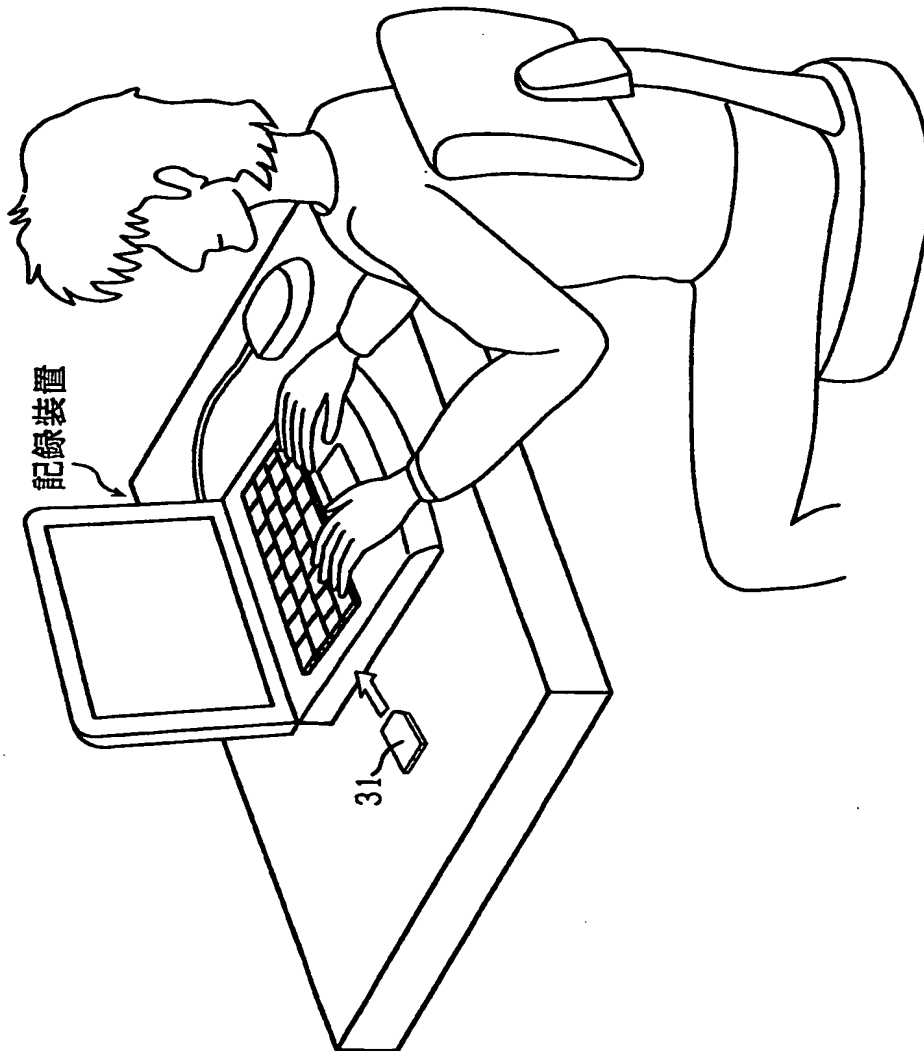
【図 6 1】



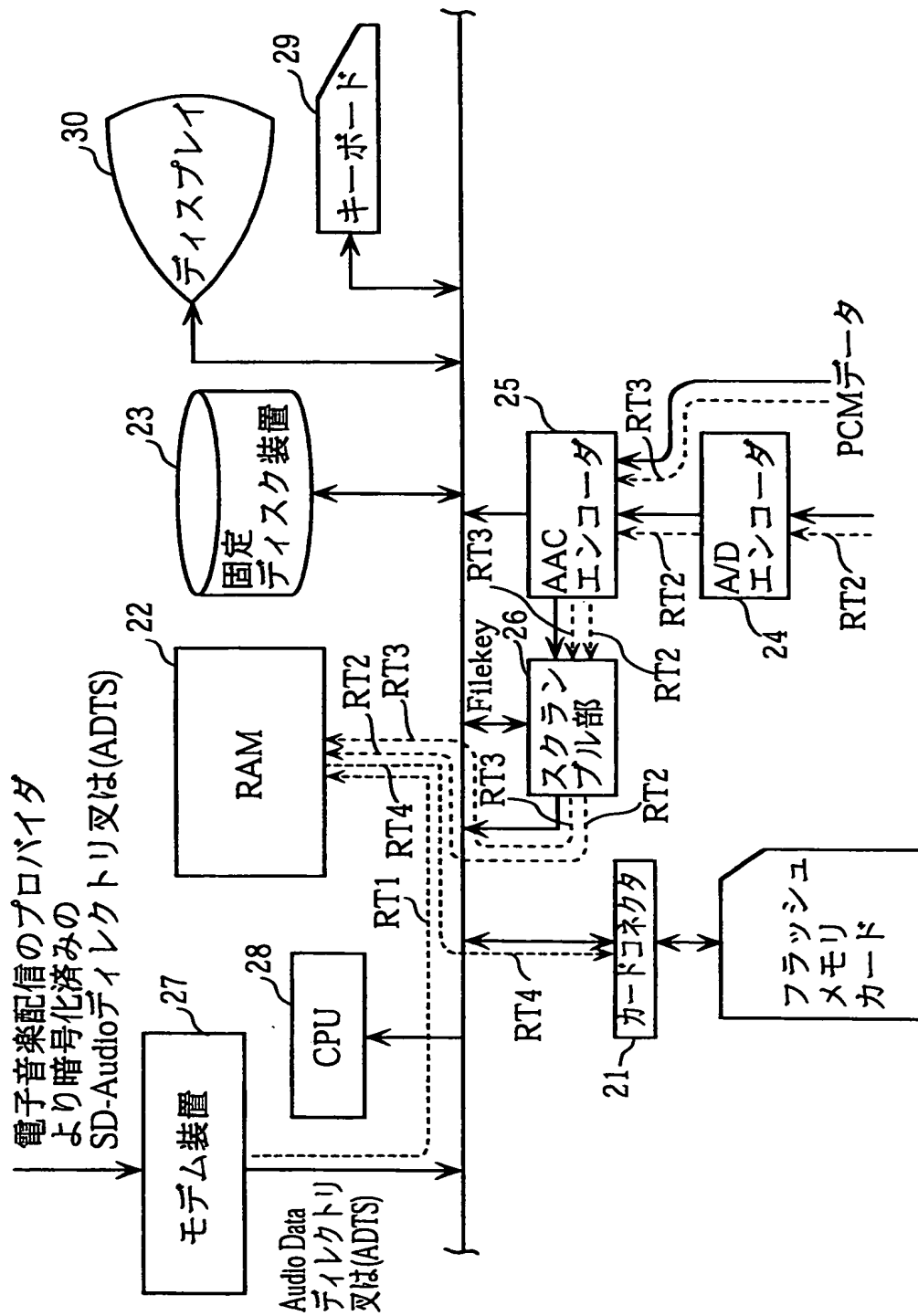
【図 6 2】



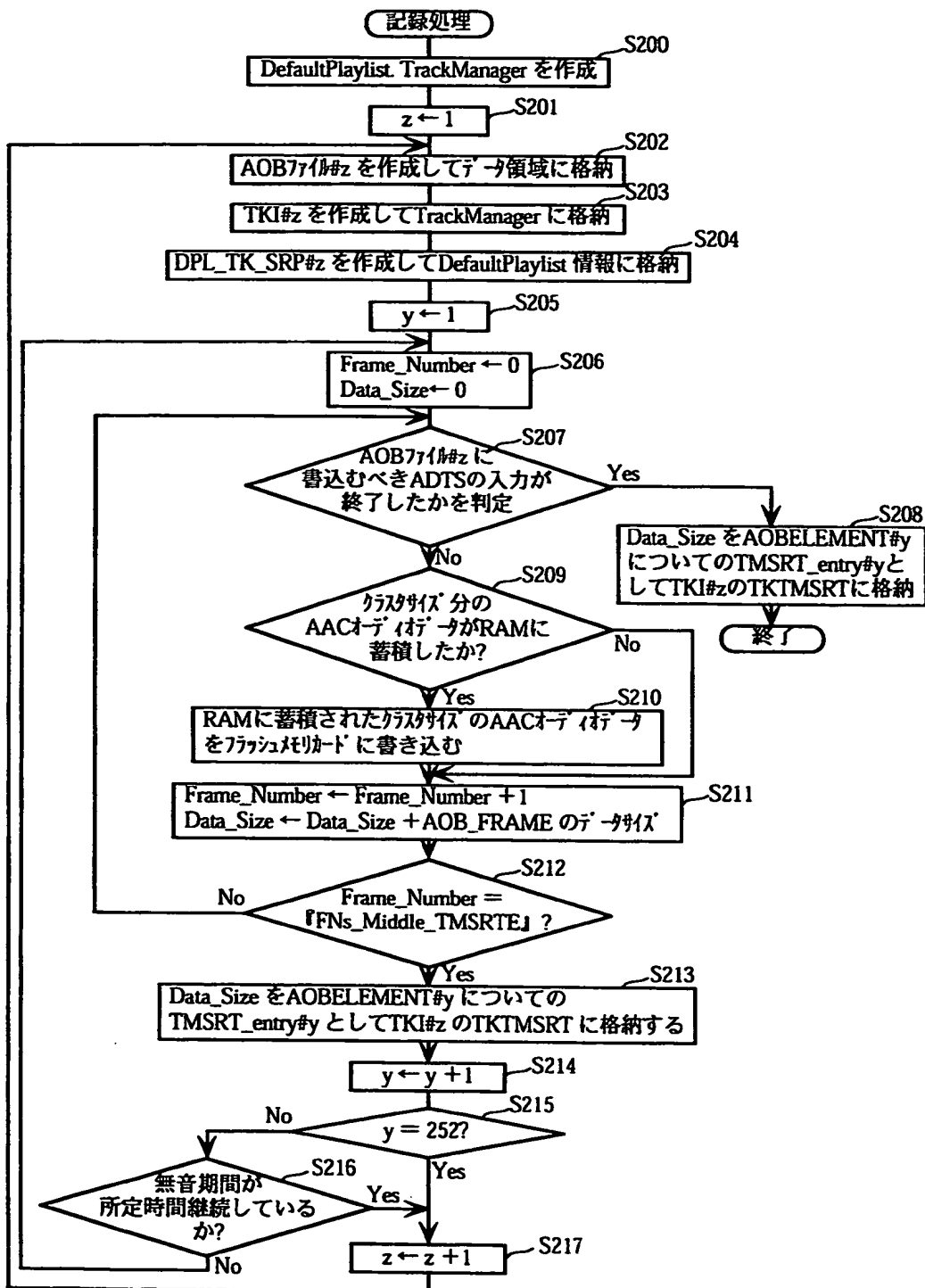
【図 63】



【図 64】



【図 6 5】



【書類名】 要約書

【要約】

【課題】 音楽コンテンツの著作権を保護しつつも、音楽コンテンツの編集を行うことができる半導体メモリカードを提供する。

【解決手段】

複数のAOBファイルは、オーディオストリームを分割することにより得られた複数のAOBを収録しており、互いに異なる暗号鍵にて暗号化されている。

トラック管理情報(TKI)は、各ファイルに対応づけられており、その対応づけられたファイルに収録されているオーディオブロックが、音楽著作物におけるどのような曲として管理すべきか又は電子本著作物におけるどのような章として管理すべきかを示す。プレイリスト情報(Default_Playlist情報)は、各トラック管理情報及びファイルのそれぞれを指し示す複数のサーチポインタ(DPL_TK_SRP)からなり、各サーチポインタの並びにより、曲若しくは章として管理される複数のオーディオブロックをどのような順序で再生させるかを示す。

【選択図】 図 3 7

出 願 人 履 歴 情 報

識別番号 [000005821]

1. 変更年月日 1990年 8月28日

[変更理由] 新規登録

住 所 大阪府門真市大字門真1006番地

氏 名 松下電器産業株式会社